VACCINE

Visual Analytics for Command, Control and Interoperability Environments
A U.S. Department of Homeland Security
Science and Technology Center of Excellence

VACCINE ANNUAL REPORT – YEAR 6
Addendum A - Publications
JULY 1, 2014 – JUNE 30, 2015
Cooperative Agreement No. 2009-ST-061-CI0001

PURDUE UNIVERSITY

Georgia Tech
SFU SIMON FRASER UNIVERSITY THINKING OF THE WORLD
Stanford University
UNIVERSITY of WASHINGTON
UNCCHARLOTTE
MORGAN STATE UNIVERSITY
OAK RIDGE National Laboratory
ASU ARIZONA STATE UNIVERSITY
Carleton UNIVERSITY
VICTORIA UNIVERSITY
UNIVERSITY OF CALGARY
DALHOUSIE UNIVERSITY Inspiring Minds
UNIVERSITY OF MANITOBA
Universität Stuttgart
UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY
PRAIRIE VIEW A&M UNIVERSITY
UNIVERSITY OF NOTRE DAME
UNIVERSITY OF OXFORD
THE UNIVERSITY OF TEXAS AT AUSTIN
Virginia Tech
UNIVERSITY OF MARYLAND

U.S. DEPARTMENT OF HOMELAND SECURITY

HOMELAND SECURITY UNIVERSITY PROGRAMS
TODAY'S RESEARCH & EDUCATION, TOMORROW'S SECURITY

Business Intelligence from Social Media: A Study from the VAST Box Office Challenge

Yafeng Lu, Feng Wang, and Ross Maciejewski, *Member, IEEE*

Abstract—With over 16 million Tweets per hour, 600 new blogs posts per minute and 400 million active users on Facebook, businesses have begun searching for ways to turn real-time consumer based posts into actionable intelligence. The goal is to extract information from this noisy, unstructured data and use it for trend analysis and prediction. Current practices support the notion visual analytics can play a large role in enabling the effective analysis of such data. However, empirical evidence demonstrating the effectiveness of a visual analytics solution is still lacking. This paper presents a visual analytics system which extracts data from Bitly and Twitter to use for box office revenue and user rating predictions. Results from the VAST Box Office Challenge 2013 demonstrate the benefit of an interactive environment for predictive analysis compared to a purely statistical modeling approach. These visual analysis method used in our system can be generalized to other domain where social media data is involved, such as sales forecasting, advertisement analysis, etc.

Index Terms—social media, box office, visualization, prediction

1 Introduction

Social media data presents a promising, albeit challenging, source of data for business intelligence. Customers voluntarily discuss products and companies, giving a real-time pulse of brand sentiment and adoption. Unfortunately, such data is noisy and unstructured, making it difficult to easily extract real-time intelligence. Thus, the use of such data can be time-consuming and cost prohibitive for businesses. One promising current direction is the application of visual analytics. Recently, the visual analytics community has begun focusing on the extraction of knowledge from unstructured social media data [12]. Studies have ranged from geo-temporal anomaly detection [3], [4] to topic extraction [14] to customer sentiment analysis [5]. The development of such tools now enables end-users to explore this rich source of information and mine it for business intelligence.

One key area for business intelligence is revenue prediction. One means of revenue prediction is utilizing social media to understand product adoption and sentiment. Currently, very few tools exist that effectively enable the exploration of social media (such as Twitter) in conjunction with traditional business intelligence analytics (such as linear regression). Due to the abundance of social media discussions on movies, movie revenue prediction has drawn much attention from both the movie industry and academic field. Movie meta-data, social media data and google search volumes have all been explored in various prediction methods. For example, an early study by Simonoff et al. [13] predicted box office revenue with a logged response regression model using meta data features (e.g., time of year , genre, MPAA rating) as categorical regressors. Zhang et al., [15] demonstrated that regression models based on meta data features can be enhanced by utilizing variables extracted from news sources, and Joshi et al. [6] explored the relationship between film critic reviews and box office performance. Further work by Asur et al. [1] found that the rate of Tweets per day could explain nearly 80% of the variance in movie revenue prediction, and recent work from Google [10] claimed a 94% prediction accuracy in box office prediction by utilizing the volume of internet trailer searches for a given movie title.

While such methods have demonstrated the benefits of social media for extracting business intelligence for box office revenue prediction, they have relied solely on automated extraction and knowledge prediction. This paper presents our visual analytics toolkit for movie box office prediction. Our toolkit consists of a web-deployable series of linked visualization views that combine statistical techniques (multiple linear regression and time series modeling) with data mining (sentiment analysis) for predicting the opening weekend gross and viewer rating scores of upcoming movies. This type of visual analytics approach for social media analysis and forecasting can be directly applied to a wide range of business intelligence problems. Understanding how information is spread as well as the underlying sentiment of the messages being spread can give analysts critical insight into the general "pulse" of their brand or product. Developing a set of quick look visualization tools for an overview of such social media data along and linking this to models that business analysts generate for deploying new products, advertising campaigns and

- *Yafeng Lu, Feng Wang, and Ross Maciejewski, are with the School of Computing, Informatics and Decision Systems Engineering at Arizona State Unviersity.*
  E-mail: {lyafeng, fwang49, rmacieje}@asu.edu.
- *Visual Analytics and Data Exploration Research (VADER) Lab - http://vader.lab.asu.edu*

sales forecasts can be critical. Our toolkit can also be used to explore other business related social media data, for example, to see how well an ads campaign did and the pattern of information spreading. Some exploration can help adjust business decisions.

In order to demonstrate the effectiveness of our system, this paper reports on the results of the Visual Analytics Science and Technology (VAST) Box Office Challenge 2013. Results from this challenge also allowed us to explore the hypothesis that the use of visual analytics will enable end-users to develop better box-office predictions when compared to a purely statistical solution. Our results demonstrate that our analytics team was able to outperform the purely statistical model solution during the course of this contest; however, results from this study merely support the hypothesis that visual analytics can improve an end-user's analytic capabilities. More studies are required to create further convincing evidence.

2 Data Extraction, Analysis and Visualization Tools for Box Office Predictions

In order to explore the impact that visual analytics can have on generating insight into social media data, our work focused on box-office predictions using Twitter indices, bitly links, and access to the Internet Movie Database. This system is a web-enabled visual analytics toolkit that allows analysts to quickly extract, visualize and clean information from social media sources. These tools were combined with linear regression and temporal modeling for movie box office prediction and sentiment analysis for movie review rating prediction. In this section, we will discuss the various tools developed as well as lessons learned from the contest.

2.1 Tweet Mining – Overview, Sentiment and Cleaning Tools

While the tools developed are applicable to a variety of social media analysis problems, our specific application focused on structured data from the internet movie database (e.g., genre, budget, rating), and unstructured data from social media (e.g., Tweets, blog posts). While structured data is relatively straightforward to extract, unstructured data requires a large amount of pre-processing and manipulation. Unstructured data collected from social media revolved around movie related Tweets and bitly URLS. Tweets were collected for the two-week period prior to the release date based off the hashtag provided by a movie's official Twitter account. Our goal was to develop tools that could extract a variety of metrics from Twitter and IMDB (see the summary in Table 1 of the metrics we found most useful). Several of the extracted metrics required data mining and cleaning. To facilitate this, we developed tools that could present the volume of Tweets at various levels of temporal aggregation(Figure 1 (a)), enable users to remove unrelated Tweets from the aggregate metrics, and allow users to extract and manually adjust the sentiment of a Tweet (Figure 1 (b-d)).

TABLE 1: Variables Description

Variable	Description
OW	3-day Opening Weekend Gross
Budget	Approximate movie budget from IMDB. (unit is "million" of dollars)
Genre(category)	The movie's genre(s) according to IMDB
TUser	Number of unique users who tweeted about a movie
TBD	The average daily number of Tweets over the 2 weeks prior to release
TSS	Tweet Sentiment Score - A summation of each individual word's sentiment polarity as calculated via SentiWordNet [2]
MSS	Movie Sentiment Score - A derivation of the overall sentiment of a movie
MSP	Movie Star Power - A summation of the Twitter followers of the three highest billed movie stars (as listed by IMDB)

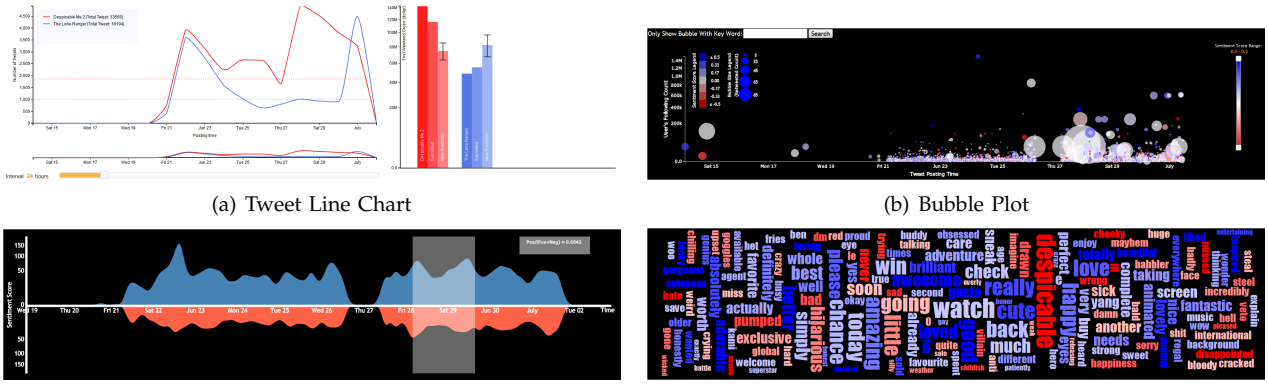
In order to approximate the popular sentiment of a movie, we processed each Tweet using a dictionary based classifier, SentiWordNet [2]. This process assigns each word in the Tweet with a score from -1 to 1 with -1 being the highest negative sentiment score and 1 being the highest positive sentiment score. Next, each Tweet is assigned a sentiment score by summing the sentiment score of all words in the Tweet and scaling the range from $-.5$ to $.5$ (TSS in Table 1). Finally, the movie sentiment score (MSS in Table 1) is calculated as

$$MSS = \frac{Positive\ Score}{Positive\ Score + Negative\ Score} \quad (1)$$

where *Positive Score* is the sum of all Tweets for a given movie with a TSS greater than zero and *Negative Score* is the absolute value of the sum of all Tweets for a given movie with a TSS less than zero.

Once the sentiment scores for Tweets were extracted, these values were then visualized to the end user. Figure 1 (b-d) shows the bubble plot view, the sentiment river view, and the sentiment wordle view. In the sentiment wordle view (Figure 1 (d)), the 200 most frequently mentioned words are extracted and visualized.

Both the bubble plot and the wordle plot enabled interactive searching and filtering by keywords and users. Users posting irrelevant messages could be removed from the Tweet count and mismatched sentiment could be modified by the end user. The primary use we found for the views in Figure 1 were for data cleaning. The primary lesson learned was that visualization tools are a necessity for data cleaning due to the noisiness of social media data and the problems inherent in sentiment matching using a sentiment dictionary (e.g., phrases such as "I want to see this movie so bad" are marked as negative due to the word "bad", and words such as "Despicable" give negative sentiment when they are merely references to a movie title). While the wordle view provided a quick way to assess the sentiment of popular words, it was necessary to hover over the bubble



(a) Tweet Line Chart

(b) Bubble Plot

(c) Sentiment River

(d) Sentiment Wordle

Fig. 1: Tweet trend and sentiment views for Despicable Me 2. (a) Line charts and bar graphs showing the number of Tweets per day and the predictions. (b) A Tweet bubble plot where blue represents positive sentiment and red represents negative. The size of the bubble represents the number of times a Tweet has been retweeted, the x-axis is time, and the y-axis is the number of followers that the user who submitted the Tweet has. (c) A sentiment river view where sentiment is aggregated over four hour intervals. Positive sentiment is plotted in red above the x-axis, negative in blue below. A user can select an area on the river to see the ratio of positive to negative sentiment. (d)A sentiment wordle where the size of the word represents the number of times it was used in a Tweet and color represents sentiment. By clicking a word, the bubble chart view will be filtered to only Tweets containing that word.

plot or open a Tweet list view through the search bar in order to fully explore the context of a Tweet. While such views were useful for data cleaning, our analysis approach (see Section 3) demonstrated to us that these views were more effective for cleaning and overview than for use in the model analysis. The critical need for tools to extract the correct metrics for regression modeling is a major hurdle that needs to be overcome in utilizing social media data for business intelligence. The bubble plot and wordle plot helped us to deal with the challenge of sentiment analysis and cleaning of noise from social media data.

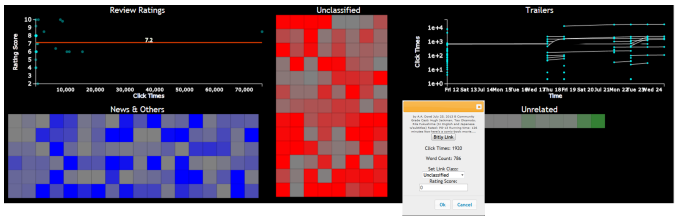


Fig. 2: Our interactive bitly classification widget. In the center are the unclassified links which the user can click and classify as seen in the floating window. The upper left is a plot of review score by click counts with a line for the average review score value.

2.2 Bitly Mining

While Tweets could be reasonably processed via the SentiWordNet dictionary, blog posts required a different approach. As part of this work, we explored long-form text by extracting bitly links containing movie keywords. These links typically consisted of review articles or news reports about the movies (or in many cases unrelated news, for example when the movie "The Heat" was released, the Miami basketball team, The Heat, had just won the NBA championship). For our review score prediction, we relied on prescreening review scores that were embedded in bitly links and developed an interactive tool for extracting these scores as shown in Figure 2. Initially, each bitly link starts as unclassified and is represented in a pixel matrix (color saturation corresponds to the number of times a link was clicked). By clicking on an unclassified square, a pop-up box appears with a brief bit of text from the article. The user can then choose to follow the link to scan the article for review scores and then manually assign a review score to an article or classify it as news or unrelated. A plot of review scores from articles versus the number of times an article was accessed is provided for analysis (see the upper left quadrant of Figure 2). This tool allows for quick data filtering and extraction, for example, reviews for the Star Trek video game can easily be separated from the Star Trek movie which would be difficult to automatically encode. Furthermore, the color coding from the pixel matrix can be used as a metric for classifying only those articles that had a substantial amount of views.

Similar to our lessons learned in Tweet mining, extracting information from bitly can be difficult to fully automate. As in the Star Trek example, multiple products for a movie may be released and reviewed at the same time. Furthermore, review scores may range from "two thumbs up" to "4 out of 5 stars" to "6 out of 10". With the analyst in the loop, these scores can be mapped to a user's own base system (in this case our metric was out

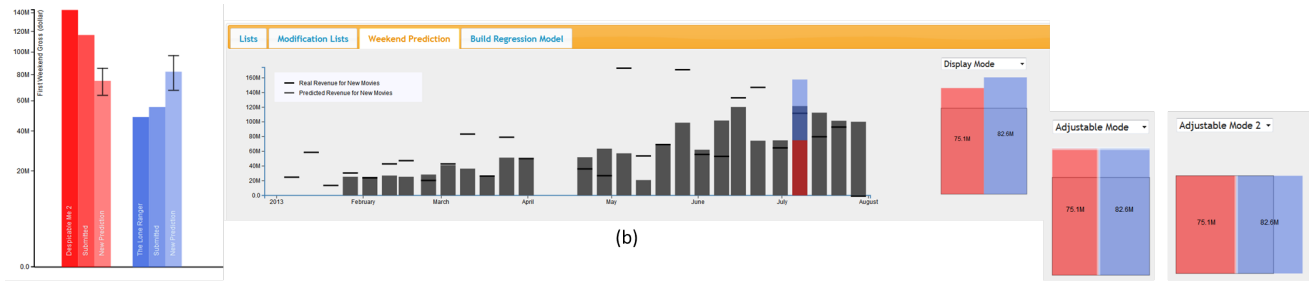


Fig. 3: The weekend prediction view for newly released movies and the prediction adjustment widget. This is the weekend when Despicable Me 2 and The Lone Ranger were released. (a) The bar graph view showing the actual value, submitted prediction and model prediction. (b) The stacked graph view showing the predicted weekend gross overlaid with the upcoming movie's regression model prediction. (c) The adjustment widget where users can modify the gross prediction; however, the predicted values for the new movies remain proportional. (d) The adjustment widget for changing individual predictions. The gray box represents the total weekend gross.

of 10).

2.3 Regression Modeling

Once data cleaning and variable extraction was complete, the next task was to use the social media metrics to develop a model for predicting box office revenue and review scores. Traditional variables used in these box office prediction models include structured variables (e.g., MPAA rating,movie budget) and derived measures (e.g., popularity of the movie stars, popular sentiment regarding the movie). Based on our initial literature search, we chose to utilize multiple linear regression for an initial prediction range for the opening weekend box office revenue (see the sidebar for a brief introduction to multiple linear regression modeling). We explored a variety of different variables that could be mined from the contest, see Table 1. After initial model fitting and evaluation using R [9], we found our best fit to be of the form:

$$OW = \beta_0 + \beta_1 TBD + \beta_2 Budget + \varepsilon \qquad (2)$$

The model is updated weekly as new movies are entered into the data set. Parameters are fit using movie data beginning in January, 2013. Our first prediction was for the May 17th weekend and used data from 39 movies for training. Our weekly models reported an $R\text{-}adj^2 \approx 0.60$ with $p < .05$. Our final parameters were $\beta_0 \approx 4.9 \times 10^3$, $\beta_1 \approx 4462$, and $\beta_2 \approx 2.3 \times 10^5$.

The drawback of this model is that it does not fit the data overly well and predictions have a large variance. For comparison, a linear regression model using google search volumes was reported to explain more than 90% of the variance on box office performance [10], and models by Asur et al. [1]also report an $R\text{-}adj^2$ of over 90% when the number of theaters was used as a regressor. Our hypothesis was that a visual analytics toolkit could partially enable analysts to overcome poor data (partially due to the noise in social media data and partially due to the closed world nature of the contest). In order to facilitate better model prediction, we created a simple bar graph view (Figure 3(a)) which, for historical movies, showed the model prediction and its 95% confidence interval error range, our submitted prediction, and the actual box office gross. For new movies, only the model prediction and user submission was shown. This view was critical in our analysis process, and the primary view into the data consists of an overview of the Tweets per day and the model predictions of the movies under analysis as shown in Figure 1(a).

2.4 Temporal Modeling

While the regression model is able to provide one point for analysis, our goal was to also provide a big picture overview. For any given weekend, there is likely a maximum amount of money available in the market. In order to approximate the total amount of money available in the market, we employed a simple moving average model. Limitations here included access to data (historical weekend grosses were not available, and after a movie opens, further weekend takes were no longer reported in the contest). To compensate for this, we approximated subsequent weekend grosses for movies under the assumption that movies would run for three weeks following their opening weekend, and each weekend their box office take would be reduced by 50%. Thus, for any given weekend, we approximated the gross as:

$$\textit{Weekend Gross}(t) = \sum_{\forall_i} \mathrm{OW}_i(t) + \sum_{\forall_i, j=1}^{j=3} .5^j \mathrm{OW}_i(t-j),$$

where t is the current weekend and i is the index to a movie that exists at time t. Then, for the weekend gross prediction, we use a moving average:

$$\textit{Weekend Gross}(t+1) = \frac{1}{3} \sum_{j=0}^{j=2} \textit{Weekend Gross}(t-j).$$

Finally, we approximate the available revenue for new movies as:

$$\textit{New Movie Gross}(t+1) = \textit{Weekend Gross}(t+1) - \sum_{\forall_i, j=1}^{j=3} .5^j \mathrm{OW}_i(t+1-j).$$

While this prediction is crude, it provided the analysts with a valuable bound in which to explore the revenue predictions.

Results from the temporal weekend prediction and the linear regression models were then visualized in two different views as shown in Figure 3. The first view consists of a linked bar graph combined with stacked bars as shown in Figure 3 (b). The primary portion of the bar graph consists of light gray bars indicating the predicted total weekend market for the new movies and the dark gray short line indicates the actual weekend market for each calendar week whose date is shown on the x-axis. The stacked color bar graph is visualized only for the weekend under analysis, and the color design is the same as the movie's color in the prediction bar graph.

The second view, Figure 3 (c) and (d), is used to enable users to interactively adjust predictions while also visualizing the bounds of the total weekend prediction. In this view, a gray square is drawn, the area of which is scaled linearly to the total weekend prediction. Colored rectangles are superimposed onto the gray square, where the area of each colored rectangle represents the linear regression prediction for each movie being released on that weekend. If the sum of the individual predictions is equal to the total prediction, the colored rectangles will fit exactly into the gray square in both Figure 3 (c) and Figure 3 (d). The color design is the same as those of the bar graph, and modifying the size of a bar in any view will modify the size across all views.

Our system was designed to allow for three types of prediction adjustments.

1) Users are allowed to change the amount of the total gross prediction but the ratio between the movies will remain consistent.
2) Users are allowed to change the amount of an individual prediction but the total weekend prediction is kept consistent.
3) Users are allowed to arbitrarily change each movie's prediction and ignore the weekend gross.

By implementing and integrating multiple comparison methods, we found that we were able to quickly bound our analysis. While flexible, these bounds provided us with an early estimate of the total expected weekend gross in which to compare the predictions of our linear regression models. This multiple model comparison was a critical step for our overall box office prediction and was regularly used for all movie analyses.

While the results of our temporal predictions were of low quality, the combination of predictions and bounding of the problem space provided critical information for comparison and analysis. We will further discuss in Section 3 how the combination of both models was critical for successful predictions. Overall, the addition of multiple models predicting similar information can help guide analysts to a better ground truth. Similar to principles employed in the delphi method [11], where predictions are solicited from multiple experts and used to come to a common conclusion, in our system, we allow users to solicit predictions from multiple models to aid in their analysis. This bounded adjustment widget can be used in other hierarchical predictions which have both individual and total predictions, such as sub-topic trend prediction in a time period.

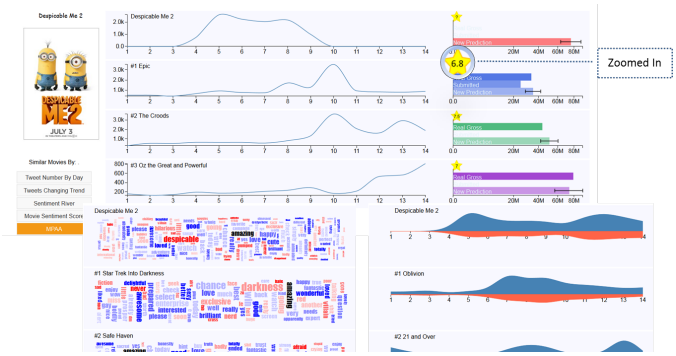


Fig. 4: A user defined similarity view cropped to show the topmost similar movies. In the center is the Tweets by day view, on the right is a graph of the opening weekend gross. There are bars for the actual gross, our final prediction, and the prediction range. The star in the upper left corner of the graph shows the review score.

2.5 Similarity Visualization

While bounding the movie predictions provided context for an overview of the total weekend, our other critical analytic view was the similarity widget. This widget enables analysts to quickly find and compare the accuracy of prediction based on various criteria of similarity. This allows analysts to determine if the given prediction model typically underestimates, overestimates or is relatively accurate with regards to movies that the analyst deems to be similar. In this manner, a user can further refine their final prediction value for both the box office gross and the review score. In this work, we have defined nine similarity criteria with distance calculation methods defined in Table 2. In all similarity matches, we show the top five most similar movies. These views allow users to directly compare Tweet trends and sentiment words between movies deemed to be similar in a category. Figure 4 contains snapshots from the Despicable Me 2 similarity page showing the line chart view with an MPAA similarity criterion, a wordle view with top word similarity criteria and a theme river view with sentiment similarity criteria.

While all of the variables used in our similarity metric could also be used in the linear regression model, results of the modeling indicated that these variables were not significant in altering the model. However, by providing an analyst with insight into these secondary variables,

TABLE 2: Calculations of Similarity Criteria

Similarity Criteria	Distance Measurement
Tweet Number by Day	$Dis(v,s) = \sum_{i=1}^{14} \lvert TBD_i(v) - TBD_i(s) \rvert$
Tweet Changing Trend	$Dis(v,s) = \sum_{i=1}^{14} \left\lvert \frac{TBD\_i(v)}{Max(TBD\_j(v), j=1,2,...14)} - \frac{TBD\_i(s)}{Max(TBD\_j(s), j=1,2,...14)} \right\rvert$
Sentiment River	$Dis(v,s) = \sum_{i=1}^{14} \left\lvert \frac{MSS\_i(v)}{Max(MSS\_j(v), j=1,2,...14)} - \frac{MSS\_i(s)}{Max(MSS\_j(s), j=1,2,...14)} \right\rvert$
MSS	$Dis(v,s) = \lvert MSS(v) - MSS(s) \rvert$
MPAA	same MPAA rating and close release date
Genre	$Dis(v,s) = 1 - \frac{card(Genre(v) \cap Genre(s)) \times 2}{card(Genre(v)) + card(Genre(s))}$
MSP	$Dis(v,s) = \lvert MSP(v) - MSP(s) \rvert$
Sentiment Wordle	$Dis(v,s) = 1 - \frac{card(SWordle(v) \cap SWordle(s))}{card(SWordle(v))}$

coupled with the temporal weekend modeling, further refinement of the prediction is made possible. For example, an analyst may compare the absolute difference between Tweets of two movies, or they can inspect the trend of the Tweets through line chart comparison using the Tweets Changing Trend similarity metric. This tool also allows users to quickly compare the current movies under analysis to recently released movies with the same Motion Picture Association of America rating, genre or movie stars popularity based on the number of Twitter followers a star has.

3 A Visual Analytics Process for Box Office Predictions

This system was used to predict 23 movies over the course of 3 months in the VAST 2013 Box Office Challenge. Our prediction process involved 3 steps. Our example prediction process focuses on the July 4th holiday in the United States when Despicable Me 2 and The Lone Ranger were released.

3.1 Movie Review Score Prediction Process

Our movie review score process centered around using the wisdom of the crowd for predicting an expected IMDB review score. For each movie, our process began by entering the bitly view and manually extracting review scores from bitly users who had done a prescreening of the movie (Figure 2). In the case of Despicable Me 2, the analysts manually classified the most clicked bitly reviews. The average value of all review scores extracted for Despicable Me 2 was 7.8. Once the average value is recorded we would then use the similarity view to compare to other movies. The movie review score is visualized as a star highlighting the review value in the corner of the bar graphs (Figure 4). Typically we would compare across genre, movie rating and sentiment to determine if we felt the average value extracted from bitly links was a reasonable prediction. In the case of Despicable Me 2, we compared to Monsters University as both movies were animated sequels. Monsters Universitys IMDB rating was 7.8 giving us confidence that our predicted value of 7.8 was reasonable. This same process was then performed for the Lone Ranger, and a viewer rating of 6.4 was predicted.

3.2 Movie Gross Prediction Process

Once the viewer rating was predicted, we then focused on determining the box office gross for the two movies. This weekend was challenging for two reasons. First, the data stream from the contest was broken, providing only 6 days worth of Tweets, and, second, the predictions were for a five-day weekend as opposed to the typical three-day weekend. Using the available data, we obtained a rough estimate for the Despicable Me 2 box office value in the range of $76M +/- $13M and $85M +/- $13M for The Lone Ranger. Next, we explore the expected three-day weekend total and see that our time series model approximates that $124M is available for the two movies for the three-day weekend. A quick look at Figure 3 shows that our regression predictions are well outside the bounds of the time series model prediction.

Given the misalignment between the two models, we begin exploring the similarity views to determine which movies The Lone Ranger and Despicable Me 2 are most similar to based on our predicted review score as well as various other metrics. We compare Despicable Me 2 to a variety of animated movies and we see that the predicted $73M is actually low when compared to animated movies such as Monsters University. Next, we explore various similarity views for The Lone Ranger and see that it is likely similar to World War Z, which had a weekend gross of $66M. However, World War Z's viewer rating was much higher at 7.4 than the predicted 6.4 for The Lone Ranger.

After looking at the available information, we determined that Despicable Me 2 should perform similarly to Monsters University, and we predicted a three-day gross of $85M. Based on our temporal prediction, this left only $39M for The Lone Ranger; however, given the other evidence, it seemed likely that The Lone Ranger would underperform. Finally, we took our three-day prediction values and linearly scaled them to be a five day prediction, resulting in a final five day prediction of $116.5M for Despicable Me 2 and $55.45M for The Lone Ranger. The actual three-day gross for Despicable Me 2 was $83.5M and $29M for The Lone Ranger. The actual five-day gross for Despicable Me 2 was $143M and $48.7M for The Lone Ranger, and the actual IMDB ratings were 7.9 for Despicable Me 2 and 6.8 for The Lone Ranger.

TABLE 3: Comparison with Peer Teams Predictions

Team	Gross Prediction				Viewer Rating			
	Entry	Average Error	STD	MRAE	Entry	Average Error	STD	MRAE
VADER(Interactive)	23	11.213	9.416	0.467	23	0.487	0.460	0.075
Team Prolix	23	16.466	15.195	0.424	20	0.82	0.640	0.129
Uni Konstanz Boxoffice	14	17.056	15.743	3.929	21	0.905	1.519	0.095
CinemAviz	21	17.219	17.677	1.970	21	0.738	0.559	0.114
Team Turboknopf	8	21.9	15.606	0.685	18	0.514	0.426	0.079
elvertoncf - UFMG	3	12.677	9.806	3.009	3	1.323	0.328	0.259
Philipp Omentisch	5	30.657	38.028	0.678	5	0.5	0.324	0.071
CDE IIIT	2	60.6	62.084	0.537	2	0	0	0

4 Results From VAST Challenge

Eight teams (our team being Team VADER) from various research institutes participated in the VAST Box Office Challenge. Data was also collected from 4 professional movie prediction websites. In this section, we compare our prediction performance with respect to peer teams from the VAST challenge and professional predictions.

4.1 Comparison with Peer Teams

Table 3 provides summary statistics of the performance of each team that participated in the VAST Box Office Challenge. For the gross prediction we report the average error (in terms of millions of dollars), the standard deviation (STD) of the average error term and the mean relative absolute error (MRAE), which is the percentage of bias deviating from the real value.

$$MRAE = \frac{1}{N}\sum_{i=1}^{N}\frac{|Prediction_i - RealValue_i|}{RealValue_i} \quad (3)$$

Similar values are reported with regards to predicting the IMDB rating (in the case of the IMDB rating, participants submitted a rating score from 1-10). These statistics can be interpreted by their magnitude, where smaller values indicate more accurate predictions. Data collected in Table 3 was provided to all challenge participants after the contest was closed.

In terms of average error and standard deviation, our team reported the lowest values in gross prediction across all teams. With respect to the MRAE for gross prediction and viewer rating, our results are slightly worse than Team Prolix (MRAE of .424 for Prolix compared with our .467), and similar in range to Philipp Omentisch, CDE IIIT and Team Turboknopf. While Team Prolix was able to achieve a smaller MRAE over the contest than our group, comparatively, they have a much larger average error and standard deviation indicating more inconsistency in their predictions.

With regards to the viewer rating prediction, our team had the lowest average error and MRAE of all teams with more than 5 submissions. CDE IIIT submitted two perfect predictions; however, those were CDE IIIT's only predictions making it difficult to determine if their methods would produce consistent results. With regards to the average error and standard deviation of the viewer rating, our team had similar results to Team Turboknopf, slightly besting them with regards to Average Error, but being slightly worse with regards to standard deviation.

TABLE 4: Comparison with Professional Predictions.

Prediction Source	Entry	Average Error	STD	Average MRAE
VADER (interactive)	21	12.729	9.425	0.285
VADER (No interaction)	21	23.051	22.011	0.501
boxoffice.com	21	8.538	7.466	0.191
filmgo.net	6	12.75	7.409	0.297
hsx	20	9.06	7.397	0.205
boxofficemojo	14	9.864	7.527	0.224

4.2 Comparison with Professional Predictions

In order to explore the hypothesis that the use of visual analytics will enable end-users to develop better box-office predictions we have also collected results from four professional prediction websites for comparison. For our comparison to the professional prediction websites, we again explore the results of the VAST Box Office challenge. Given that these results were collected and verified by the contest organizers, we feel this is an adequate means of justifying their validity. For the comparison in Table 4, only 21 movies are shown in the chart as two movies, The Bling Ring and The To Do List, were limited release movies which opened in only 5 and 591 theaters respectively and most expert prediction sites do not provide predictions for limited release movies. For each prediction, we followed the same general process as described in Section 3. As previously stated, the underlying linear regression model used in our system was significant with an R^2-$adj \approx .6$.

Results in terms of the MRAE are given in Figures 5 and 6 for the opening weekend gross and review score respectively. Figure 5 provides a comparison of our MRAE with that of several expert prediction websites. From Figure 5, it is clear that we outperformed the experts in the case of three movies (Epic, Hangover 3 and Fast and Furious 6), and in the case where we had the largest error (After Earth) we relied heavily on the analytical component with no interaction.

Table 4 gives the average error, standard deviation and MRAE for the predicted movies. What the results show is that for the model used, the predictions of our team utilizing an interactive tool were a dramatic

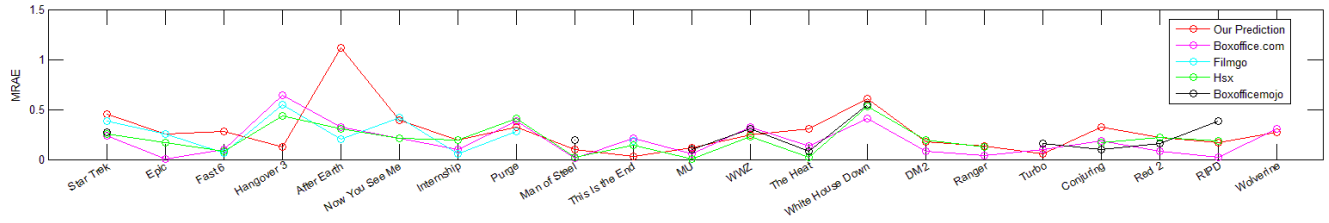


Fig. 5: The mean relative absolute error of box office weekend gross predictions, where the x-axis is the predicted movies.

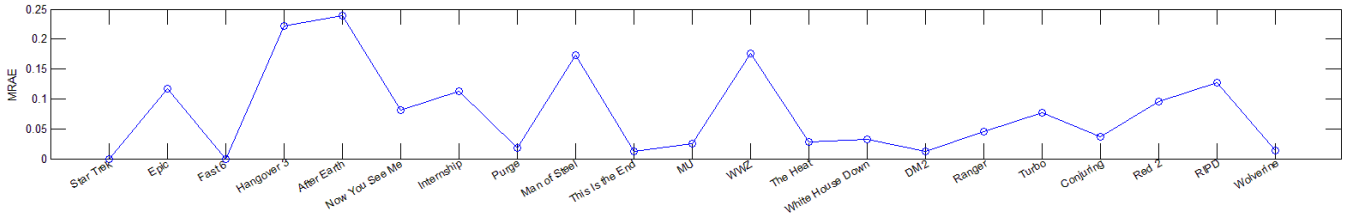


Fig. 6: The mean relative absolute error of our viewer rating predictions, where the x-axis is the predicted movies.

improvement over just the model itself (see Table 4 VADER (Interactive) versus VADER (No Interaction)). This provides a strong indication that the hypothesis that the use of visual analytics will enable end-users to develop better box-office predictions when compared to a purely statistical solution is valid. However, we do not wish to overstate our claims. This contest provides only a single data point for exploring how one group of analysts in a closed world setting were able to utilize a visual analytics toolkit for improved prediction. What this demonstrates is the need for further controlled studies in which a group of analysts perform similar model predictions and results are compared between analysts using a visual analytics platform and analysts using only results from a given regression model. However, results from the contest indicate that a visual analytics toolkit can enhance business intelligence.

Further analysis of the data also indicates that these tools enabled our team of novice box office analysts to quickly close the gap between the experts. Table 4 shows the average error and standard deviation for our predictions and compares them to four well known professional prediction websites. What we see is that both our average error and average MRAE are slightly lower than filmgo.net indicating that our methodology enabled our group of novice analysts to be competitive when compared to expert analysts. The significance of this relies on three major assumptions:

1) The professional prediction websites have more experience in box office prediction than our team.
2) The professional prediction websites have access to more data than our team was allowed in the closed world contest.
3) Access to more data can enable better predictive models as evidenced by [1], [6], [10], [15]

First, it seems reasonable that a professional prediction website would have much more experience than a computer science team who has never previously attempted to predict box office sales. Second, it is clear that utilizing data sources (specifically the number of theaters a movie is released in) will result in a better prediction model (a larger R^2). From these assumptions, it becomes clear that (in this instance) the application of a visual analytics toolkit can enable individuals that are knowledgeable with respect to data analysis to quickly understand information being presented to them in new domains and make predictions that are in line with expert predictions. Overall, our prediction error (.285) was slightly lower than that of filmgo (.297), but approximately 50% worse than boxoffice.com (.191). However, if we remove the After Earth and Now You See Me weekend (during which we relied heavily on the model and very little on the interactive visuals), our MRAE drops to .239 which puts us near the prediction range of boxofficemojo. Other sources of error can be accounted for in disrupted Twitter and bitly data feeds. These interruptions were pronounced for The Heat, White House Down, Monsters University and World War Z. However, even with those interruptions, our predictive analysis process was still quite robust with only The Heat being a significantly worse prediction than the professional sites.

5 Conclusions and Future Work

Overall, the application of visual analytics for social media analysis has proven relatively effective. However, there are still many challenges in applying this to all domains of business intelligence. First, social media data is extremely noisy. Movie predictions work well as one can track the effectiveness of ad campaigns by following the specific hashtags promoted by a brand. As

the analysis gets farther afield from Twitter (for example when trying to mine data from bitly) it becomes difficult to choose effective keywords. Second, due to the ever changing stream of social media sources and users, it is likely that any automated system for data collection and prediction will eventually be steered off course. As such, it is critical to link the human into the loop; however, as is evidenced by the issues in sentiment analysis, the data cleaning process should not overburden the analyst. The sentiment analysis and cleaning process employed in this work places an overly large burden on the end user. As such, integrating a system for having a user label a subset of tweets for sentiment model training could be a more effective solution. Third, it is imperative to link highly curated small datasets with this so call "big data". While social media data can be used as a proxy for many signals, we find that linking multiple data sources with varying levels of reliability (for example, total weekend take for all movies and regression modeling) can enhance the predictive abilities of a system. For example, doing focus groups and linking their data with results from social media could enhance the analysis of a proposed new product release. Finally, this paper demonstrates the need for interactive tools to mine social media data. From the examples of box office prediction, it is clear that such data contains a wealth of information. However, extracting knowledge from this data and effectively communicating this remains a challenge. There are clear needs for effective data cleaning tools to improve filtering of unrelated social media signals, as well as for improving the results of challenging analytical problems (such as sentiment analysis). Our results demonstrate that the use of visual analytics tools can have a significant impact on knowledge discovery for business intelligence.

While our results are able to only demonstrate a single data point, we feel this is significant in that the provisions of the contest allow us to directly compare a group of analysts using a visual analytics toolkit to experts in a particular modeling domain. However, we recognize that this is a far cry from definitively validating the hypothesis that the use of visual analytics will enable end-users to develop better box-office predictions when compared to a purely statistical solution. Overall, this work points for the need of better methods for evaluating the impact of visual analytics when used for complex problems such as prediction. There are a variety of factors and variables that need to be addressed and controlled, including the level of expertise and the types of visualizations provided. With our current system in place, we have been collecting streaming movie data in a manner similar to the VAST Box Office Challenge and plan to run a variety of controlled experiments. Of primary interest are exploring levels of expertise and the impact that visual analytics has on resultant predictions. We feel that results shown in this paper provide an important starting point for such explorations.

6 Sidebar: Linear Regression Model Construction and Evaluation

Regression analysis is one of the most widely used methods of pattern detection and multifactor analysis [7]. With a proper regression model, data can be better described, interpreted, and predicted.

6.1 Linear Regression Model

The basic form of a k-variable linear regression model is defined as follows:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \varepsilon \tag{4}$$

Variable y is known as the response, variables $x_i, i = 1, \ldots, k$ are the regressors and ε represents the error term. The goal is to define a relationship between the response term and the regressors by solving for the linear coefficients, β_i that best map the regressors to the response. The linear regression model is most often written in matrix form such that:

$$Y = X\boldsymbol{\beta} + \varepsilon$$

$$Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} X = \begin{pmatrix} 1 & x_{11} & \ldots & x_{1k} \\ 1 & x_{21} & \ldots & x_{2k} \\ \vdots & \vdots & & \vdots \\ 1 & x_{n1} & \ldots & x_{nk} \end{pmatrix} \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{pmatrix}$$

For multiple regression models, higher order terms may also be used to model the response (e.g., 2^{nd} order variables are of the form x_i^2 and $x_i x_j$). However, for this work our focus is on the simple linear regression model.

6.2 Parameter Estimation

In order to solve for the parameters β_i the ordinary least square (OLS) solution is most commonly employed. Note that this assumes normality for the data; however, if this assumption is not valid a maximum likelihood estimation would then be employed (which is equivalent to OLS under the assumption of normality).

For OLS, we wish to minimize

$$S(\boldsymbol{\beta}) = \sum_{i=1}^{n} \varepsilon_i^2 = \boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon} = (y - X\boldsymbol{\beta})^T (y - X\boldsymbol{\beta})$$

by satisfying

$$\frac{\partial S}{\partial \boldsymbol{\beta}} \mid_{\hat{\boldsymbol{\beta}}} = -2X^T y + 2X^T X\hat{\boldsymbol{\beta}} = \mathbf{0}.$$

Under assumptions of normality, the solution takes the form of $\hat{\boldsymbol{\beta}} = (X^T X)^{-1} X^T Y$ and the prediction function is $Y = HY$ where $H = X(X^T X)^{-1} X^T$. In one-order multiple linear regression, the predicted response is a linear combination of observations.

6.3 Model Selection

In a multiple variable dataset with a single response variable (such as in our box office gross prediction), analysts will traditionally be faced with a large set of potential linear regression models consisting of various regressors and orders. For example, in box office prediction, the response could be related to the number of Tweets per

day, or the number of theaters the movie is released in, or any combination of variables.

In order to decide which model should be used in prediction, there are several principles an analyst will typically consider.

- Do not violate the scientific principle, if there exists one, behind the dataset.
- Maintain a sense of parsimony to keep the order of the model as low as possible and the number of regressors as small as possible.
- Keep an eye on extrapolation. Regression fits data in a given regressor space but there is no guarantee that the same model also applies to other data outside this space.
- Always check evaluation plots more than the statistics. Residual plots and normal plots help show outliers and lack of fit.

In order to verify the efficacy of a model, analysts will typically rely on a variety of statistical graphics to determine the critical variables in the model, i.e., those that explain the most variation with the simplest form [8]. Several statistics are usually reported to evaluate the effective fit of a given model: p-value, R^2 and R^2-adj. The p-value shows the significance of a regression model, where $p < .05$ indicates the model is significant with a 95% confidence interval. R^2 and R^2-adj generally describe the percentage of variance explained by a given model. R^2-adj specifically takes the degree of freedom into consideration and should be used in multiple regression to compensate for the increased variance when adding regressors. A model is typically selected when it has a small p-value and a high R^2 or R^2-adj value and a relatively simple form with reasonable residual distributions.

ACKNOWLEDGMENTS

This work was supported in part by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0001. The authors would like to thank the VAST challenge organizers and participants for their help in data collection, evaluation and discussions.

REFERENCES

[1] S. Asur and B. A. Huberman. Predicting the future with social media. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, pages 492–499, 2010.

[2] S. Baccianella, A. Esuli, and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the International Conference on Language Resources and Evaluation*, 2010.

[3] H. Bosch, D. Thom, F. Heimerl, E. Puttmann, S. Koch, R. Kruger, M. Worner, and T. Ertl. Scatterblogs2: Real-time monitoring of microblog messages through user-guided filtering. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2022–2031, 2013.

[4] J. Chae, D. Thom, H. Bosch, Y. Jang, R. Maciejewski, D. Ebert, and T. Ertl. Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In *IEEE Conference onVisual Analytics Science and Technology (VAST)*, pages 143–152, 2012.

[5] M. C. Hao, C. Rohrdantz, H. Janetzko, D. A. Keim, U. Dayal, L.-E. Haug, M. Hsu, and F. Stoffel. Visual sentiment analysis of customer feedback streams using geo-temporal term associations. *Information Visualization*, 2013.

[6] M. Joshi, D. Das, K. Gimpel, and N. A. Smith. Movie reviews and revenues: An experiment in text regression. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 293–296, 2010.

[7] D. C. Montgomery, E. A. Peck, and G. G. Vining. *Introduction to Linear Regression Analysis*. Wiley, 2012.

[8] T. Muhlbacher and H. Piringer. A partition-based framework for building and validating regression models. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):1962–1971, 2013.

[9] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, 2008.

[10] A. C. Reggie Panaligan. Quantifying movie magic with google search. *Google Whitepaper — Industry Perspectives + User Insights*, 2013.

[11] G. Rowe and G. Wright. The delphi technique as a forecasting tool: Issues and analysis. *International journal of forecasting*, 15(4):353–375, 1999.

[12] T. Schreck and D. Keim. Visual analysis of social media data. *Computer*, 46(5):68–75, 2013.

[13] J. S. Simonoff and I. R. Sparrow. Predicting movie grosses: Winners and losers, blockbusters and sleepers. *Chance*, 13(3):15–24, 2000.

[14] X. Wang, W. Dou, Z. Ma, J. Villalobos, Y. Chen, T. Kraft, and W. Ribarsky. I-si: Scalable architecture for analyzing latent topical-level information from social media data. *Computer Graphics Forum*, 31(3pt4):1275–1284, June 2012.

[15] W. Zhang and S. Skiena. Improving movie gross prediction through news analysis. In *Proceedings of the IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, pages 301–304, 2009.

Integrating Predictive Analytics and Social Media

Yafeng Lu, Robert Krüger, *Student Member, IEEE*, Dennis Thom, Feng Wang,
Steffen Koch, *Member, IEEE*, Thomas Ertl, *Member, IEEE*, and Ross Maciejewski, *Member, IEEE*

Abstract— A key analytical task across many domains is model building and exploration for predictive analysis. Data is collected, parsed and analyzed for relationships, and features are selected and mapped to estimate the response of a system under exploration. As social media data has grown more abundant, data can be captured that may potentially represent behavioral patterns in society. In turn, this unstructured social media data can be parsed and integrated as a key factor for predictive intelligence. In this paper, we present a framework for the development of predictive models utilizing social media data. We combine feature selection mechanisms, similarity comparisons and model cross-validation through a variety of interactive visualizations to support analysts in model building and prediction. In order to explore how predictions might be performed in such a framework, we present results from a user study focusing on social media data as a predictor for movie box-office success.

Index Terms—Social Media, Predictive Analytics, Feature Selection

1 Introduction

Research on social media has intensified in the past few years as it is seen as a means of garnering insight into human behaviors. The unstructured nature of social media data also provides unique challenges and opportunities for researchers across a variety of disciplines. Businesses are tapping into social media as a rich source of information for product design, relations management and marketing. Scientists utilize social media data as a platform for developing new algorithms for text mining (e.g., [13]) and sentiment analysis (e.g., [45]) and focus on social media as a sensor network for natural experimentation for exploring social interactions and their implications (e.g.,[47]).

In using social media as a sensor network, researchers have developed methods that capture online chatters about real world events as a means of predictive model building. For example, work by Culotta [12] explored the use of Twitter for predicting seasonal influenza. Tumasjan et al. [43] found that the magnitude of Twitter messages was strongly correlated to German elections. Eysenbach [15] utilized regression modeling of Tweet counts to predict paper citations, and Zhang et al. [48] explored mining Twitter for emotions and predicting the opening-value of the stock market.

Currently, the visual analytics community has begun focusing on social media analytics with respect to developing tools and frameworks to collect, monitor, analyze and visualize social media data. Studies have ranged from geo-temporal anomaly detection (e.g., [9]) to topic extraction (e.g., [46]) to customer sentiment analysis (e.g., [33]). Such work focuses on capturing the incoming streams and enables the analysts to perform exploratory data analysis. However, little work has been done on developing tools for predictive analytics using social media. In 2013, the Visual Analytics Science and Technology (VAST) conference ran the VAST Box Office challenge using social media data to predict the opening weekend gross of movies. This particular contest served as an entry point to explore how users interact with visualization tools to develop predictions. Continuing from this contest, our work has focused on utilizing movie data from social media to explore the promises and pitfalls of visualization for predictive analytics. Unlike more specialized data sources (e.g., criminal incident reports, emergency department data, traffic data, etc.), movie data lends itself well to analyzing visual analytics modules as many casual users think of themselves as movie domain experts.

In this paper, we present a framework for social media integration, analysis and prediction. This framework consists of tools for extracting, analyzing and modeling trends across various social media platforms. In order to test our framework, we focus on the specific problem of predicting the opening weekend box-office gross of upcoming movies. This system integrates unstructured data from Twitter and YouTube with curated data from the Internet Movie Database (IMDB). Temporal trends and sentiment are extracted and visualized from social media, and IMDB features can be explored through parallel coordinate plots. Specifically, this tool was developed to support the exploration of predictive models while integrating user interaction to iteratively update the models, compare against past models, and explore similarities between movies. To demonstrate the efficacy of our system, we tested our framework with seven subjects and evaluated their prediction performance. We present lessons learned and future directions for improving the user in the loop workflow for predictive analytics.

- *Yafeng Lu, Feng Wang, and Ross Maciejewski, are with Arizona State University. E-mail: {lyafeng, fwang49, rmacieje,}@asu.edu.*
- *Robert Krueger, Dennis Thom, Steffen Koch, and Thomas Ertl are with University of Stuttgart, Germany. E-mail:{Robert.Krüger, Dennis.Thom, Steffen.Koch, thomas.ertl}@vis.uni-stuttgart.de.*

Manuscript received 31 Mar. 2014; accepted 1 Aug. 2014; date of publication xx xxx 2014; date of current version xx xxx 2014.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

2 Related Work

This paper focuses on enabling analysts to explore, validate and filter social media data for predictive analytics. In this section, we discuss past work on current state-of-the-art in visual analytics surrounding both social media data and predictive model development.

2.1 Visual Analytics of Social Media Data

Recent visual analytics systems for social media analysis include Whisper [8], which focused on information propagation in Twitter, SensePlace2 [28], which focused on the analysis of geographically weighted Tweets, and TweetXplorer [31] which combined geographical visualization of Tweets along with their social networks. Other applications have explored the use of social media analytics for improving situational awareness in emergency response. Thom et al. [42] and Chae et al. [9] developed spatiotemporal visual analytics systems that integrated various social media data sources for anomaly event detection and disaster management. Our proposed framework takes cues from this previous work and is developed to integrate data from multiple sources, for our case study, we integrate Twitter, YouTube and IMDB data.

A wide variety of work also exists with regards to social media topic extraction and sentiment analysis of social media. Dou et al. [13] developed an algorithm for hierarchically organizing news content based on topic modeling. Hao et al. [18] applied topic based stream analysis techniques to detect sentiment in Tweets and created a sentiment cal-

Fig. 1: Front Page of the Frozen Weekend. View (a) is the Tweet and Youtube comments line. The solid lines indicate the number of Tweets per day starting 14 days before the release (x-axis). The left y-axis indicates the number of Tweets.The dashed lines represents the number of Youtube comments per day using the right y-axis. Each color represents one movie. Clicking the legend highlights the corresponding trend line. View (b) is the opening weekend gross bar graph. The left bar indicates the real gross while the right bar indicates the baseline model's prediction. View (c) shows the Tweets and users.

endar and map. Nguyuen et al. [33] applied machine learning to Twitter to extract sentiment and compare dictionary based and machine-learning sentiment classifiers. Wang et al. [45] created a sentiment analysis and visualization system called SentiView to analyze public sentiment in Tweets and BlogPosts. Similar to previous work [24, 27], our framework also performs sentiment analysis on the ingested social media data. However, while previous work relies directly on automatic algorithms, we allow the users to interactively modify the sentiment of an item (e.g., a Tweet) as a means of correcting for classification errors. Overall, our framework builds upon prior visual analytics work with regards to social media analytics and expands this domain with regards to integrating predictive analysis and model building tools.

2.2 Predictive Analytics

It is important to note that our proposed framework is not the first to address predictive analytics. A variety of solutions exist for both novice and expert users (e.g., R [37], SAS [39], Weka [17], JMP [36], Excel). These software packages and tools provide a variety of machine learning algorithms that can be used for predictive analytics tasks, such as feature selection, parameter optimization and result validation. Many of these systems offer basic visualizations including residual plots, scatterplots and linecharts. However, most of their visualization are only used to display the final results and do not provide interactive means for manipulation, feature selection or model refinement; instead, these systems often opt to show baseline models or simple statistical measures for result validation, working as more of a black-box system. The goal of our framework is to directly integrate the analyst into the model building loop by enabling feature selection for model building and comparison. We include tools such as Parallel Coordinate Plots [21] and correlation rankings for quick comparison. Moreover, we have also created a variety of mechanisms for automatically suggesting similar instances within a dataset to enable the analyst to identify outliers and validate models based on the accuracy of prediction with regards to similar instances.

Recently, researchers in the visual analytics community have been developing methods for improving model building and predictive analytics. Berger et al. [5] used regression models for parameter space exploration. Choo et. al. [10] provided a classification system, iVisClassifier, using linear discriminant analysis to reduce dimensionality for improved data classification. Brown et al. [7] designed an interactive visual analysis system to improve clustering results by updating the distance function based on users' feedback to the display. We also integrate feature selection and sample filtering, but our system does not require users to be familiar with specific prediction algorithms. Instead, we focus on how much information and manipulation should be open to the user [2].

Most closely related to our work is that of Mühlbacher et al. [32] which developed an interactive visual framework for selecting subset features to improve regression models. They used R^2 to rank 1D features and 2D feature pairs, as well as a partition-based feature ranking. Their goal is to approximate the local distribution of a given target, and their visual analysis method helps to select subset features for regression models and validate the quality of models. Similar to their measure of selecting features, we also use a goodness-of-fit measure. Furthermore, we allow users to explore the correlation between features by using Parallel Coordinate Plots (PCP) because a good subset of features should also avoid multicorrelation [30]. Mühlbacher et al. also provides two general partitioning methods: domain-uniform and frequency-uniform. In our framework, local pattern detection is provided through brushing data items on any dimension from the PCP. We also allow users to choose to only train on brushed data items. Thus local patterns can be indicated by the goodness-of-fit of the model.

Since we enable users to select different features and training sets, we also allow for multiple model creation and comparison. This is akin to the Delphi method [34, 38] which has multiple experts forecast and modify their prediction iteratively by comparing to other experts' predictions before finalizing their results. In general, the Delphi method is used to obtain the most reliable consensus of group opinions. Our predictive analytics framework uses the concept from the Delphi method to allow users to make their prediction after building and exploring multiple models in multiple rounds. Similar to the Delphi method, in our system the user evaluates results, where each model represents one

expert or one round of the expert's prediction.

3 Framework for Predictive Social Analytics

Our framework focuses on integrating multi-source data from social media for analysis and prediction. We combine trend analysis, sentiment analysis, similarity metrics and feature selection for model building, evaluation and prediction. In order to evaluate this framework, we deploy our tools to the problem of weekend box-office prediction. We combine data from IMDB, Twitter and YouTube and explore this data across a variety of visual analytics modalities. The system was built using D3 [6], JSON, R [37] and WEKA [17]. The use of R and WEKA allowed for direct integration of multivariate regression and support vector machines, while D3 was used to create charts and graphics for the interactive visualization. A client-server architecture was chosen in order to allow easy portability and testing of the system across platforms, and we also explored the use of Amazon cloud services. We used the Jersey RESTful web service [22] to enable the interface between the web interface and backend server. Preprocessing was done for sentiment analysis and word frequency counts and nearly-interactive rates are obtained for visualizing the data described below. By nearly-interactive, we mean that if the data is cached, the visualizations can be updated at greater than 10 frames per second (FPS), if the data is not cached then the user will see a wait symbol and typically experience a 5 second lag on the first query, after which the exploration of that movie's features will be at interactive frame rates.

3.1 Data Description

In data representation and exploration, we focused on views for social media data sources, such as Twitter and Youtube. As Twitter data is unstructured and dirty, it requires a deeper preprocessing and manipulation before extracting high quality features.

Twitter: We collected Tweets for 112 movies released since 2013. Tweets are collected based on the hashtag posted by a movie's official Twitter account. In all we have 2.5 million Tweets and each Tweet includes the posting time, retweet status, user profile information and Tweet text sentiment.

Youtube: We used a rule-based script to collect Youtube data which contains the total viewcount and timestamps. We then calculate a range of features such as comment volume and interpolated view counts prior to the opening weekend. Overall we were able to collect about 7 million YouTube comments for 1104 movies.

The Internet Movie Database: The Internet Movie Database (IMDB) has more than 2.8 million entries (Mar. 25, 2014) with each entry consisting of hundreds of features [1]. To deal with data noise and incompleteness, the available raw text IMDB data files were first converted into an SQL-database using JMDB [44]. The data then undergoes a data cleaning procedure.Challenges include the data sparseness and huge number of nominal values, such as cast names, which hamper machine learning. To overcome the data sparseness we calculated numeric values on a per-movie basis by aggregating gross incomes and ratings of previous movies that the cast of a new movie was involved in. Finally, we obtained a high quality movie data set of approximately 2000 movies with up to 72 features per movie.

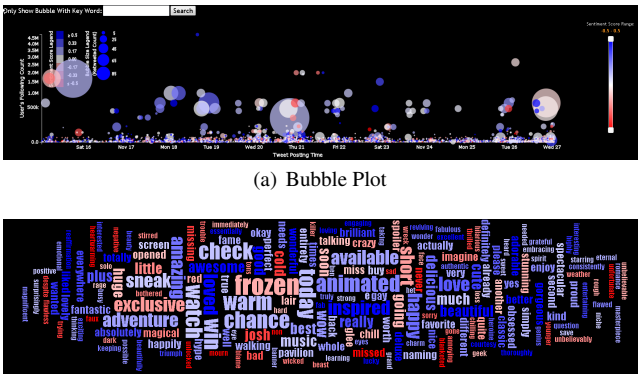
3.2 Social Media Visual Analytics

Our framework consists of a variety of views and analytical components. We provide an overview for quick trend analysis and exploration, detailed views for exploring tweet sentiment, and a similarity widget for overviews on related movies and their patterns. A core component of this framework is an iterative feature selection and model exploration module for analysis, model building and comparison.

3.2.1 Overview: Trend Analysis

When beginning analysis, users are initially presented with an overview of the data item they are trying to predict (in the case of our example, it is an overview of the movies being released in the upcoming weekend). Figure 1 shows the initial view in our web enabled system. Here, the weekend under exploration is from November 27, 2013. Figure 1(a) is a dual y-axis line chart showing the volume of Tweets and YouTube comments that have been collected relating to the movies. Users can highlight data elements by clicking on their corresponding legend entry. Key to this view is the fact that the multiple sources of data enable cross-validation. Due to the limits of the Twitter Streaming API, it is often the case that the Tweet stream will consist of missing data. However, there are many instances in which the YouTube comment traffic directly tracks that of the Twitter stream (just at different magnitudes as evidenced by the axis scales). In this manner, the analyst can quickly validate the accuracy of a source and determine what anomalies might be present.

In Figure 1(b), the user can also get an overview of a baseline linear regression model prediction for that weekend. Since data for the opening box office gross has already been collected for historical weekends, the user is also shown the actual box office value. In this manner the analyst can quickly gain insight into the limitations of a proposed model. The buttons beneath the bar charts allow the user to directly navigate to a detailed view of the movie where visualizations showing word frequency, retweet count, Twitter followers and Tweet sentiment can be explored.



(a) Bubble Plot

(b) Sentiment Wordle

Fig. 2: (a) A Tweet bubble plot where blue represents positive sentiment and red represents negative. The size of the bubble represents the number of times a Tweet has been retweeted, the x-axis is time, and the y-axis is the number of followers that Tweeter has. (b) A sentiment wordle where the word size represents the number of times it was used in a Tweet and color represents sentiment.

Figure 2 shows two of the detailed views, a temporal bubble plot and a sentiment wordle view. The bubbles in Figure 2(a) represent an individual Tweet and are colored based on the mined sentiment, where each Tweet has been processed using a dictionary based sentiment analysis, SentiWordNet [3]. This assigns each word in the Tweet to a score ranging from -1 to 1, negative to positive sentiment respectively. Each Tweet is then assigned an overall sentiment score by summing the sentiment of all words in the Tweet and then normalizing the sum. A blue color indicates positive sentiment while red indicates negative. The size of the bubble represents the number of times a Tweet was retweeted while the height on the y-axis indicates the number of followers the Twitter account has. The x-axis represents time. Similarly, all the Tweets related to a movie are converted into a wordle (Figure 2(b)), where the size of each word represents the number of times the word appears in the movie data set and the color represents the sentiment of the word. From this view, users can quickly filter for Tweets with particular keywords and they can modify the sentiment value in cases where the dictionary matching is wrong (for example, cases where the Tweet says "I want to see Frozen so bad!" will be a negative Tweet when in reality the sentiment is positive). Future work will deploy more machine learning techniques to allow for interactive

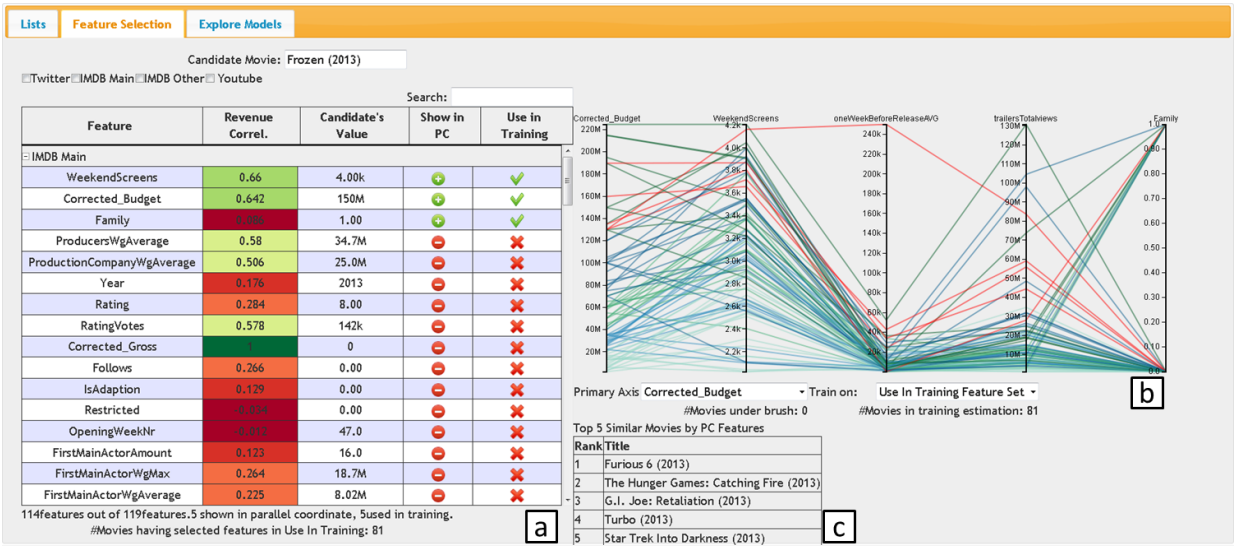


Fig. 3: Feature Selection page with Frozen as an example. View (a) is a Feature Selection table having four groups of features with their correlation to revenue mapped to a divergent color scheme. View (b) is a Parallel Coordinate view with the five most similar movies highlighted in red. View (c) lists the five most similar movies suggested by the system based on features in the PCP.

Tweet labeling for advanced sentiment classification and analysis.

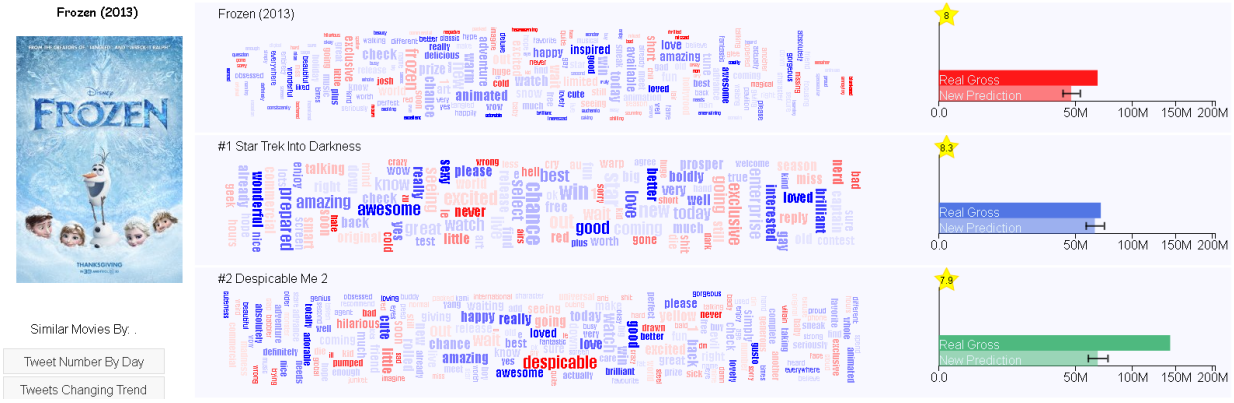
3.2.2 Feature Analysis and Selection

While the overview and detail visualizations enable exploratory data analysis, the key contribution of our work is the interactive modeling and prediction components. Feature values of movies can give insights and hints about their box office success. Moreover, they can be used as predictors for a movie's opening weekend revenue. Using Twitter, Youtube and IMDB data sources, we extracted four groups of features for model building with 119 features listed in the Feature Selection Table (Figure 3). Given the large number of features, it is necessary to provide the users with a suitable starting point for analysis. As such, we utilized known predictive features for movie analysis from previous work [41] (e.g., budget, number of screens the movie opens on, etc.). Thus, when the users begin their exploration process, they are presented with a baseline model to compare against. Other options would include integrating automatic feature selection as an entry point for analysis (e.g., [26, 49]).

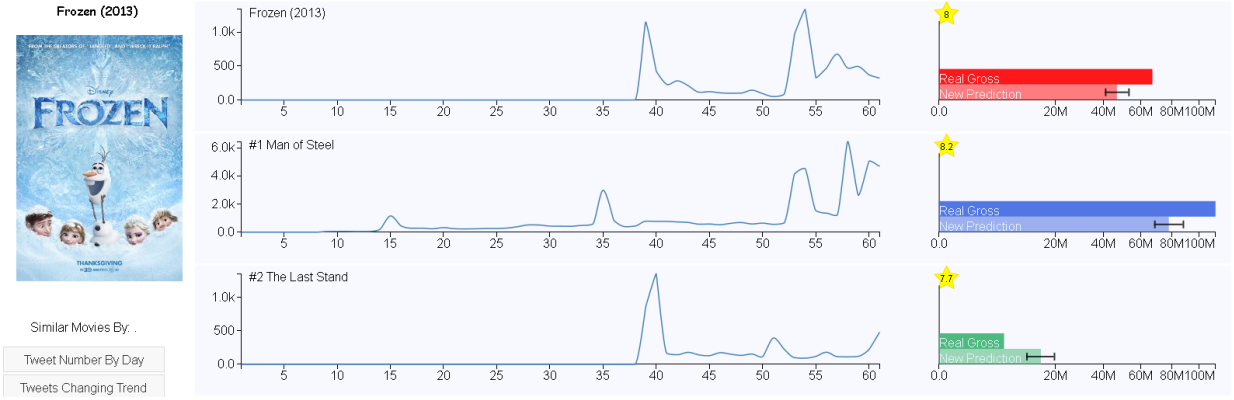
Our goal was to augment model building by adding tools for a user to modify and explore various features. In order to quickly enable this exploration, our Feature Selection Table (Figure 3(a)) utilizes a variety of interactions and visual overlays. First, for the candidate movie being predicted (in this case Frozen), features which are not available are grayed out. Second, each of the columns in the feature selection table provides the details of a movie. The first three columns include information on the feature's name, the correlation to the revenue, and the candidate movie's value. These columns can be automatically sorted from high to low or low to high simply by clicking on the column header. The Revenue Correlation column is also color coded to directly highlight correlated features. A myriad of work has been done in feature selection [29, 35, 40] and correlation is traditionally used as one of the major factors in feature selection. A high correlation of a feature to the response variable (in our case the movie revenue) indicates that this feature could greatly impact the model. We use a green to red divergent color scale [19] where green represents a high absolute value of correlation and red represents a low value of correlation, with .5 being the midpoint value. Although correlation here is univariate (meaning we do not show correlations between multiple features) and non-linear dependencies are not taken into account, it still provides important information to users for feature detection and analysis.

The final two columns in the Feature Selection Table are associated with the Parallel Coordinate plot visualization and the model training data selection. The "Show in PC" column, when selected, will add that feature as an axis of the Parallel Coordinate Plot. The "Use in Training" column, when selected, will add all data elements that contain all of the features selected into the training set. To quickly see what features have been selected, the analyst can sort the features by clicking the column header. When features are selected, the footer information about the Feature Selection Table will update and tell the user how many features have been added to the training set, as well as the amount of movies that exist having all of these features. In this manner, the analyst can determine how many data elements can be used to train a model and they can quickly make decisions about the tradeoff between the use of more features or more training samples. For example, if a user chooses to select a Twitter feature, only 112 movies in our data set have associated Twitter data. Thus, the number of elements in the training set decreases. However, Twitter data may have a high correlation to the opening weekend gross. As such, the analyst can actually build multiple models with multiple features for training and analysis.

Another way to select the training data is through interaction with the parallel coordinate plot view. Let us consider the case in which a user has sorted the features by correlation to revenue, selected some features with higher correlation to the gross, and selected features that he/she suspects are important. These selected features can now be further explored in the PCP view (Figure 3(b)) by simply activating the "Show in PC" cell in the corresponding table row. Referring to the candidate movie's value, shown in the fourth column, the user can further filter out movies far away from this value in the PCP view. Figure 3(b) shows features of the movie "Frozen" with highly correlated features in different group and the movie's genre, "Family". Pairwise correlations between features are explored in the PCP view. For example, the WeekendScreens (the number of screens in which a movie was released during its opening weekend) and the oneWeekBeforeReleeaseAVG (the daily average number of Tweets that are related to a movie one week before its opening) variables are correlated. These axes can be dragged and dropped to explore more pairwise dimension correlations so that an analyst can choose features with low multi-correlation in order to improve the model performance. Users can then interactively select ranges on each axis to filter the data and can select an option to train the model using only the selected data.



(a) Similarity by Sentiment Wordle



(b) Similarity by Youtube Trailer Trend

Fig. 4: Similarity Widget View with Frozen. (a) is the top two most similar movies' wordle view with the Tweet sentiment wordle as the similarity criteria. Each wordle consists of the top 200 sentiment words. (b) is the top two most similar movies' line chart view with the 1 year Youtube Trailer view count trend as the similarity criteria.

The PCP view can also be used to generate insight into the data. For example, by brushing and selecting only Family movies using the Boolean genre feature "Family," one can define the training set to be only those movies that are considered to be "Family" movies. Moreover, the PCP view allows the analyst to select a primary axis, this selection defines the feature on which we base the PCP line color scheme. For example, if we color the lines based on the genre axis "Family" we can see that family movies rarely obtain a very high gross. From there, the user could train the model for only Family movies or could look for genre crossover movies such as Family and Animation.

The final item in our Feature Analysis and Selection widget is the "Top 5 Similar Movies by PCP Features" view, Figure 3(c). Given the feature vector corresponding to the features selected in the parallel coordinate plot, our system automatically calculates a Euclidean distance metric between the candidate movie and all other movies that appear in the PCP view. The five movies with the smallest Euclidean distance are then summarized in a tabular view.

3.2.3 Similarity Widget

While the Feature Analysis and Selection Tools show the top 5 most similar movies, we have also developed a series of tools for enabling users to explore temporal and sentiment similarities with regards to social media trends and specific feature similarities such as genre and ratings. Figure 4 shows our similarity widget page. Items in this similarity view focus primarily on similarity across social media (as opposed to the previous widget which used a Euclidean distance metric across many features, this view is a pairwise feature similarity). The left side of Figure 4 shows the various similarity options provided while the center view displays line charts or wordles depending on the selection. We have ten predefined metrics and one "Make Your Own Similarity" option. The rightmost area shows the model predictions and the actual weekend gross for similar movies via a bar graph.

This widget enables analysts to quickly find and compare the accuracy of predictions based on various criteria of similarity, and to perceive if the given prediction model typically underestimates, overestimates or is relatively accurate with regards to movies that the analyst deems to be similar. In this manner, a user can further refine their final prediction value. In this work, we have defined ten similarity criteria with distance calculation methods focusing on matching temporal trends through sequential normalization or Euclidean distance metrics for magnitude comparisons. In all similarity matches, we show the top five most similar movies. These views allow users to directly compare Tweet trends and sentiment words between movies deemed to be similar in a category. Figure 4 contains snapshots from Frozen's similarity page cropped to the top two most similar movies by Sentiment Wordle and Youtube Trailer Comments.

Though similarity metrics used in this page are not directly transformed into modeling features, by providing an analyst with insight into these secondary variables, coupled with the model performance with similar movies included in the training set, further refinement of the prediction is made possible. For example, an analyst may compare the absolute difference between Tweets/Youtube comments of two movies, or they can inspect the trend of the Tweets through line chart comparison using the Tweets Changing Trend similarity metric. This tool also allows users to quickly compare the current movies un-

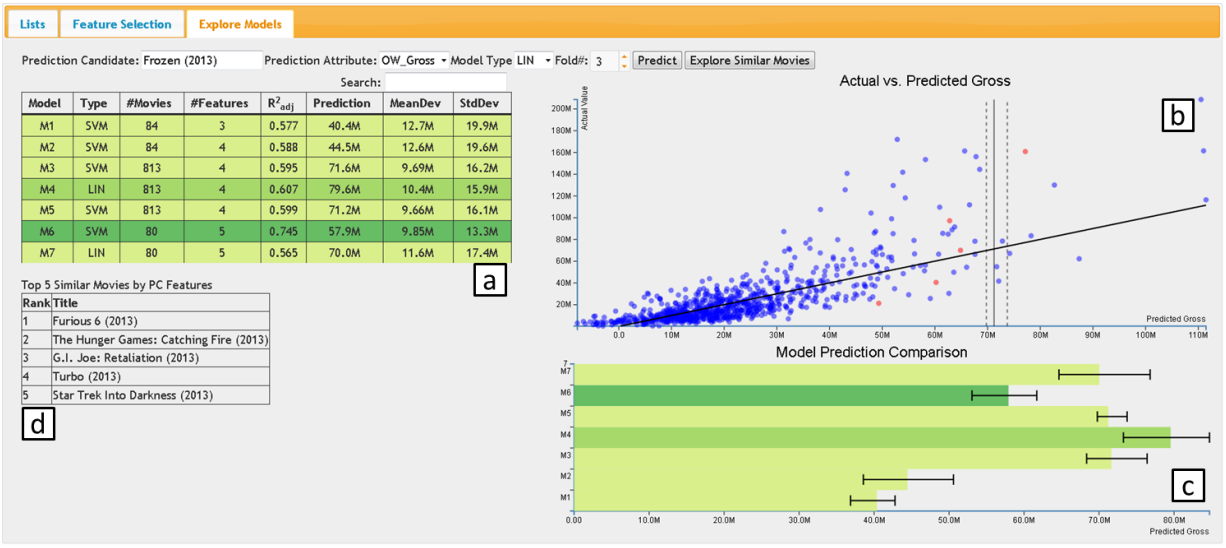


Fig. 5: Multiple Method Modeling with Frozen as the candidate movie. View (a) is a trackable Model History Table recording each model the user built. View (b) is the scatterplot of the Actual vs. Predicted Gross showing a model's prediction for each movie in the training set and the prediction result together with a stable range for the candidate movie. View (c) is the bar graph of the Model Prediction Comparison having each model's prediction stacked. View (d) lists the five most similar movies as was done in View (c) of the Feature Selection page.

der analysis to recently released movies with the same MPAA rating and genre. When the user builds a model involving Twitter features, the top 5 most similar movies listed in the Feature Selection and the Explore Models page can be compared in the similarity page.

3.3 Model Building, Analysis and Verification

Based on recent literature and the general use of prediction models, we support the creation of three different types of models: Support Vector Machine (SVM) [11], Linear Regression (LIN) [30] and Multilayer Perceptron (MLP) [20]. Using the linear regression model with the budget and the average number of daily Tweet (TBD)s for a movie as regressors and the opening weekend gross as response, the system provides users with a baseline prediction result together with a 95% confidence interval for each movie. The baseline model results are shown in both the front page (see Figure 1(b)) and the similarity page's right-hand bar graphs (Figure 4).

Besides exploring the baseline model, the user can build a more complex model, bringing in domain knowledge and analytic insights. For instance, the user is allowed to interactively set up parameters and build models with different feature sets, training instances (movies) and model types. We use several error measures to give the analyst feedback about the quality of fit and the prediction stability. By using the interactive Feature Selection and Explore Models pages, the user can iteratively change the features, training sets and model types to improve a model's quality. We measure the model's accuracy using the adjusted R^2, denoted R^2_{adj}. Using R^2_{adj} has the following advantage: R^2 never decreases when a regressor (feature) is added to the model, regardless of the value of the contribution of that variable; however, R^2_{adj} will only increase when adding a variable to the model if the addition of the variable reduces the residual mean square. Otherwise R^2_{adj} decreases when adding terms that are not helpful [30]. With a feature set of size p and a number of instances (movies) n, R^2_{adj} is defined as:

$$R^2_{adj} = 1 - \frac{SS_{Res}/(n-p)}{SS_T/(n-1)} \quad (1)$$

where SS_{Res} is the sum of squares of the residual, and SS_T is the sum of squares of total.

3.3.1 Base Line Model

We used the model proposed in our VAST Boxoffice Challenge 2013 submission [27] as our base line model, which is described as follows:

$$OW = \beta_0 + \beta_1 TBD + \beta_2 Budget + \varepsilon \quad (2)$$

With all 110 movies in the training set, the estimation of parameters in Equation 2 are $OW = 6.878 \times 10^6 + 1303 \times TBD + 0.26 \times Budget$ with $R^2_{adj} \approx 0.6$ and $P \ll 0.05$.

3.3.2 Advanced Models

As most of the attributes are proportional to the box office success (e.g. the more budget, the higher weekend gross potential) we can even achieve good results using linear regression model. More advanced models can be built using a Support Vector Machine (SVM) or a Neural Network, i.e. Multilayer Perceptron (MLP). To achieve good results, these algorithms have to be finely configured by setting input parameters based on the input data. We ran a grid search (parameter optimization method) to find out the best parameter settings. For SVM we use a linear kernel and a nu-parameter of 0.4, which constrains the influence of a single instance (movie) to the model. Considering the relatively small number of movies when compared to the large feature space we also tested an RBF kernel. However this did not achieve better R^2_{adj} results than with the linear kernel. For MLP we use the backpropagation learning rule and use a learning rate of 0.3, 200 training epochs and a momentum rate of 0.85 to achieve good results.

3.3.3 Multiple Methods Modeling

Predictive models help to reveal relationships between the predictors and the response variable, but no matter how good the prediction is, no cause-effect relationship can be implied. Also, the accuracy of one prediction can hardly be generalized to all other predictions. In statistical analysis, experts usually explore residual distributions, outliers, influential points, and model stability. In our system, besides using statistical methods, we apply visual analysis methods for exploring the residual distribution.

In the page "Explore Models" the user can select which algorithm to use, set the number of folds for the stability test, train models to predict the movie's revenue, and compare between models. The Explore Models view is shown in Figure 5. For model building, the feature

and training set configurations from the Feature Selection page are applied. After the prediction is executed, the analyst can use the Actual vs. Predicted Gross view (Figure 5(b)) to obtain an overview of the residuals, as was presented in [24]. A diagonal referential line indicating "the perfect prediction" is also drawn. This means, the closer the data points lie to the referential line, the better the overall fit of the model. The top 5 most similar movies are highlighted in red to quickly guide comparison and analysis. The user can change these similar movies based on adding/removing features in the Parallel Coordinates view (Feature Selection page). To submit a good prediction for a particular movie, it may be more important that the model fits for similar movies than fits the overall training set. In other words, if the model predicts well for similar movies this may be an indicator that it also gives good results for the prediction candidate.

Our tools also enable the exploration of influential points. An influential point is an outlier in both the predictor and the response domain, and these points are known to have a noticeable impact on the model coefficients [30]. If an influential point is removed from the training set, the fit of the model will change by a relatively large degree and usually fit other points better. This fact can be used to improve prediction results. Instead of using statistic diagnostics, such as Cook's D and DFFITS [4], we allow the user to directly remove such instances and only train on selected movies. In this way, influential points can be implicitly removed via exploring differences between different models.

Finally, the Model History Table (Figure 5(a)) enables the comparison of multiple models so that the analyst can review the predictions by re-investigating their scatterplots. In combination with the Model Comparison view (Figure 5(c)), the user can also get an overview of the prediction deviations, review the increase or decrease of prediction precision and select his/her final prediction. Our goal is to build a model which can help the analyst to better predict the upcoming movie's opening weekend gross, not to build an adequate model that fits all the training data very well.

To estimate the performance and to test the model's stability, we provide an n-fold cross-validation [16, 23]. For the cross-validation we partition the data into n folds. Each fold includes num_{movies}/n instances. The movies of each fold are predicted once, using the other folds for training. This way, we ensure that the model generalizes and is not overfit to the training data. For the prediction candidate, every fold is used once to predict the outcome. Thus, for each prediction we get n results. The dashed vertical line in the scatterplot shows the range of these results. A smaller range indicates that the model is stable. This range is also shown in the bar graph below the scatter plot, where all predictions can be compared.

3.3.4 Auxiliary Analysis

Instead of depending totally on an automatic model, most industry predictions also utilize an expert's domain knowledge. For example, if a movie is released next to an expected blockbuster, its performance could be also impacted. With our system, analysts can query any movie by its title to investigate features. Users can also go to previous weekends to see how much money those movies made. A user can also investigate the Twitter and Youtube data to explore the advertising campaign and public sentiment. Usually a successful movie has either an effective advertisement campaign, positive public reactions, or both. From the bubble plot shown in Figure 2(a), large bubbles usually are Tweets from the movie production company and the bubble size indicates the spread power. If the large bubbles separate along the time line, it is likely that the company has continued advertising its movie.

4 Case Study: Predicting Disney's Frozen

This section demonstrates how an analyst would use our system to predict Frozen's opening weekend gross. This process consists of multiple steps, which can be iteratively traversed in different ways. However, we suggest the following procedure. First, the user gets an overview of the Twitter and Youtube comments using the dual-y-axis line chart to compare movies released together. Second, details can be investigated using the detail pages of the candidate movie. Third, the user can explore similar movies and compare their gross, as well as how well the baseline model performed for them. After having a general impression of the expected revenue, the user can navigate to the Feature Selection tab to explore and select features or filter movies to create a model. Finally the user can build and explore different models and their prediction ranges in the Explore Models view. Step 4 and step 5 can be iteratively applied until the user feels they can make a confident prediction.

To illustrate these 5 steps, we will take Frozen as an example. Starting on the overview page, the line chart in Figure 1 (a) indicates that there are 4 movies released on the same weekend (Frozen, Black Nativity, Homefront, and Oldboy).We quickly see that online chatter (Tweet and YouTube comment volume) about Frozen is not dominating the other weekend movies, in fact it is trending similarly to the movie Black Nativity. This phenomenon indicates that it is unlikely that Frozen will obtain an anomalously large gross as the market will be shared by competitors.

In the second step, using the detailed view of Frozen (see Figure 2) the Tweet sentiment is analyzed. One can see frequent Tweet keywords and the sentiment polarity. Also, the retweet volumes provides information about users' interest in the movie and the advertisement campaign. For example in Figure 2 we can see that Frozen does not have a large Tweet and retweet volume compared to other blockbusters; however it does have a very positive sentiment (blueish dots). The movie sentiment score for Frozen is approximately 0.8 which is very high among all 112 movies having Twitter data.

In the third step the similarity widget is explored (see Figure 4). This reveals that movies similar to Frozen were under-predicted with the baseline model, which predicts about \$44M for Frozen. The fourth step focuses on the analysis and selection of the movies features (see Figure 3). There are two main views for feature selection: the correlation view showing relationships between a feature and the revenue, and; the relationship among features depicted in the PCP view. From our baseline model we select the number of opening screens, the budget and the weekly average of Tweet counts as an initial feature selection. This gives us a model with $R^2_{adj} \approx 0.58$ (M1 in Figure 5). To further improve the model, we add another feature, view counts of the movie's YouTube trailers, and built both an SVM and LIN model. R^2_{adj} improved to approximately 0.6 while the prediction deviations from the different folds decreased. Next, using our background knowledge, we explore the genre of this movie (in this case the genre is "Family"). While adding the Family feature to the Parallel Coordinates, we find that the gross distribution for Family movies is significantly different to most non-Family genres. Thus, for our last prediction iteration, we add the family feature to the model. We obtained an R^2_{adj} score of 0.745. Finally, we review the Model Prediction Comparison graph and decided to finalize our prediction between \$60M to \$70M based on the best performing models.

5 Evaluation

In order to evaluate the effectiveness of this framework for predictive analytics, we performed a user study. On March 20th, 2014 we enlisted seven graduate students from China, India, the United States and Germany and asked them to predict the results of four different movies. The first two movies predicted were to provide them with baseline training, the next two movies were to be released on March 21st, thus having them do an actual future prediction. The movies we had them predict included Disney's Frozen (2013) and The Hunger Games: Catching Fire (2013) (which were used for training) and Divergent (March 21, 2014) and Muppets Most Wanted (March 21, 2014) (which were the movies to be predicted). For Frozen and the Hunger Games, their weekend box office data was removed for the training exercise in order to simulate the prediction process.

Of the seven participants, six were male, one was female and all were PhD students. Prior to participation, we surveyed them about their cinema affinity and data visualization knowledge on a scale from 1-5 (with 1 being the lowest). From the seven participants four claimed to be visualization experts. Five subjects rated their movie affinity

Table 1: Results for Frozen and Hunger Games. The opening weekend gross for Frozen is $67M and for the Hunger Games it is $158M.

subject	user1	user2	user3	user4	user5	user6	user7	BoxOffice.com	BoxofficeMojo
Prediction(Frozen)	55.9	59	50	60	57.7	62.5	58	47	44.7
Abs Error	11.1	8	17	7	9.3	4.5	9	20	22.3
Prediction(Hunger Games)	71.1	135	NA	100	95.9	86	75	166	167
Abs Error	86.9	23	NA	58	62.1	72	83	8	9

Table 2: Results for Divergent and Muppets. The opening weekend gross for Divergent is $56M and for the Muppets it is $16.5M.

subject	user1	user2	user3	user4	user5	user6	user7	BoxOffice.com	BoxofficeMojo
Prediction(Divergent)	54.1	53	40	50	30.1	47.5	48	66	51
Abs Error	1.9	3	16	6	25.9	8.5	8	10	5
Prediction(Muppets)	50.6	21.5	28	15	35	21.4	20	25	22
Abs Error	34.1	5	11.5	1.5	18.5	4.9	3.5	8.5	5.5

as low (1-2), and two rated medium (3-4). Their machine learning knowledge was mostly low, with only two participants claiming a basic knowledge of machine learning and prediction related tasks (these students had all taken regression analysis and/or data mining courses, as such we feel that they can be considered to have a relatively high level of expertise in the modeling and analysis process). The two subjects that rated their movie affinity as low were those that rated their machine learning and predictive analytics knowledge as high. Thus, we have three subjects that were casual users with limited domain knowledge and limited analytics experience, two subjects that had some domain knowledge and limited analytics experience, and two subjects that had expertise in data mining and predictive analytics but limited domain knowledge.

To introduce the system, we walked through an example analysis of the movie After Earth and explained our proposed analytics process (similar to the case study in section 4). Subjects were then asked to predict Frozen and The Hunger Games. During the analysis and prediction process of these two movies, they were open to ask any questions, such as the meaning of a feature, how to use a special function of the system, and what information could help to choose proper features and improve the model performance. After they submitted their final prediction about a movie, we told them the real gross so that they could make a comparison and adjust their strategy for the next movie. After practicing with these two movies, they used the system (unaided) to predict the new movies Divergent and Muppets Most Wanted.

To get a deeper understanding of the users analysis processes this study was carried out as talk-aloud [14] session. The users were asked to speak their thoughts out loud explaining their actions. We recorded voice and system interaction by video. After the study we summarized the key results and classified them into System Usability, Social Media Exploration, Feature Selection and Model Comparison.

5.1 System Usability

Key findings here indicated more details on system design. All subjects reported ease of use and interaction with the system. Furthermore, the length of the user study demonstrated the subjects' engagement. No instructions were given on the time needed to make a prediction; however, subjects spent over 1 hour on average tuning system parameters and exploring the data. Subjects also were excited to compare their results Monday and indicated they wanted to try this again. Design issues they faced were that they wanted even more transparency in the data. As no subject was a self-rated expert in cinema (most indicating they had seen less than two movies in the past 6 months) many of the subjects wanted more information about the movie features. They suggested direct links to the IMDB pages for the movies to allow even greater detail views. Overall, the most used views were the similarity page and the feature selection page.

Subjects all started their analysis on the overview page, exploring time series trends and comparing how they felt the movies on the weekend would fare when compared to others. They typically looked at the Twitter and YouTube volumes and sentiment data. At the beginning they found it difficult to interpret those visualizations as they were unfamiliar to a user; however, by the end of the study the users were requesting more features, wanting to create difference maps of the movies to look for keyword differences in the sentiment analysis and also to identify what was being discussed differently between YouTube and Twitter. As such, it is clear that more text analysis is needed for further insight generation. A clear example of gaining insight was shown during the analysis of the movie, Divergent. No subject in our group had heard of this movie; however, when inspecting the data they saw that The Hunger Games was often referred to in context with this movie. This grounding gave them the contextual clues which they needed in order to analyze Divergent.

Negative comments focused on the disconnect between the similar movies and the users' thought process. In the Feature Selection page, users are presented with the five most similar movies with respect to the selected PCP features. This is calculated as a Euclidean distance metric, and the calculation is a black-box to the user. As such, analysts were often wary of these movies and preferred to use the "create your own similarity" option on the similarity widget page. However, this again required more domain knowledge than some users had, with many again requesting details about what genre, rating, etc. a particular movie had. Future work should include better views for multi-dimensional similarity matches and more transparency in the similarity metrics. Yet, what the process highlights is that all subjects, even those with little self-proclaimed movie knowledge, are able to bring some background knowledge into the prediction process, which could be used to add value when compared to a purely automated prediction process.

5.2 Feature Selection

All users worked with the Feature Selection table to determine which data was available for a movie and remarked on how they felt the prediction was more reliable when they knew that the data existed. Again, this indicates that transparency in the model training can improve an analyst's confidence. During the feature selection process, most users started with the baseline settings, inspected the results and then iteratively chose more features with high correlations, reinspected and then iterated again. Other users again applied their domain expertise and chose features that seemed interesting to them. For example, the user that had seen 10 movies in the theater in the past six months used his domain knowledge to select features which are not obviously highly correlated to the revenue but these features considerably improved that subject's model.

Participants who decided to add Twitter related features typically based this choice on the genre of the movie, stating that Twitter users would be interested in Divergent but not in the Muppets. One user, with a basic background knowledge in prediction tasks commented on how the Parallel Coordinate view enabled her to choose features that were independent (i.e., not multi-correlated). Other users engaged

the PCP view to filter out movies to create models based on genre or movie ratings. Overall, they spent a large amount of time exploring features and discussing what they felt these features meant. They also found it extremely helpful to see how the selection of different features impacted the amount of movies available for training.

Negative comments revolved around users' frustration in feature selection, noting that there should be a way to provide more details on what is likely to be a good feature. For the inspection of correlations, one user noted that it was hard to use the PCP view and had difficulty distinguishing the highlights. However, the users all liked the design of the framework, and commented on how it would be useful to change the domain to look at other specific problems of interest. For future work, we plan to explore how to improve the presentation of features. Obviously showing all features (in this case 116) is a huge amount of information overload; however, we also want to involve the user and allow him/her to use domain knowledge to guide the modeling and prediction process. We plan to explore several methods of automatic feature selection as a means of organizing information for visual presentation and exploration and performing user studies across various feature set visualizations in order to explore this area.

5.3 Model Comparison

As for the Feature Selection view, participants found the model comparison features extremely useful. Starting with some initial predictions, they tried to improve the model to reduce the errors. Users often focused on prequel movies (particularly during the Hunger Games prediction) and focused on developing a model that was a good fit for known prequels or known movies within a genre. One user repeated the feature inspection, selection and modeling until he was able to create a model that strongly fit to the prequel (in the case of the Hunger Games). Others tried to inspect all outliers and then made decisions based on their domain expertise regarding movie similarity. This would lead to an iterative model building and refinement loop. Users also inspected the scatterplot and would then access the similarity comparison tools to explore the impact of Twitter on the model prediction. Users noted that Twitter seemed to have an impact depending on the type of movies, and many came to the conclusion that Twitter was relevant when predicting Science Fiction movies (such as Divergent) but less relevant when predicting Family movies (such as the Muppets). Again, subjects indicated a desire for even further transparency of the inner workings of the model prediction.

5.4 Prediction Results

Table 1 and 2 show the results of our user study in both the training trial and the actual prediction trial. For the training results (Table 1), subjects were found to have a lower error than that of the experts for Frozen; however, for the Hunger Games, subjects found this very difficult to model. It is important to note that we went through the example of After Earth, Frozen and the Hunger Games for training in order to give subjects examples of a low outlier, a good fit, and a high outlier respectively. In this way they can explore all possible scenarios prior to the actual prediction task.

For the actual results (Table 2), 5 of our 7 subjects were able to best BoxOffice.com predictions for Divergent and 2 of our 7 subjects were able to best both expert prediction websites. Only two subjects erred on the far low end of the spectrum for this movie (subjects 3 and 5). For the Muppets, 4 of our 7 subjects were able to best the experts, with one subject (subject 4) accurately predicting this would be a box office failure. Again, subject 5 was an outlier, and subject 1 predicted that the Muppets would be an outlier on the positive end of the spectrum.

Overall, the results of our study are quite positive. Given our subjects self-reported lack of movie knowledge, it is clear that the integration of social media and visual analytics for model building and prediction can quickly generate insight at a near professional prediction level. Subjects 2 and 7 had the highest self-reported domain knowledge and (as seen in Table 2) outperformed experts from BoxOffice.com (and Subject 2 outperformed the BoxofficeMojo results as well). The machine learning and regression experts were subjects 4 and 6 and they also outperformed the experts. The remaining subjects can all be considered more casual users and had a higher variability. In both future prediction cases, over half the subjects were able to best the experts over the course of a one hour training session. Furthermore, such work indicates that visual analytics can have a direct impact on the modeling and prediction process. As noted by Lazer et al. [25], there is a need for tools that can improve insight into large data analytics and an increased transparency can potentially lead to improved model efficacy. Future work will look at doing a more formal evaluation where a larger subject pool is recruited and more analysis between the three groups is performed.

6 Conclusions

This paper presents an interactive framework integrating social media and predictive analytics, and the presentation of a talk aloud study that discusses design successes, pitfalls and potential future directions. Analysts can utilize the system to explore and combine information, and underlying mechanisms for similarity matching and data filtering can help a user quickly engage in exploratory data analysis as part of the model building process. We allow for the quick integration of structured and unstructured data sources, focusing on box office predictions as our example domain. In comparison to state-of-the-art in visual analytics, we have worked towards improving a user's understanding of the modeling and training process. Our results were validated through case studies and user studies. We have demonstrated that such a tool can quickly enable non-domain experts to be competitive with domain experts in a given area. This seems to stem from a combination of a user's (in our case limited) domain knowledge with the interactive visualization interface. While with our system semi-professionals are not always able to beat the expert models from boxoffice.com and boxofficemojo, respectable results were obtained across a majority of users. As the industry's models and predictive practices are not available, it is difficult to comment on their workflow. However, talking with experts from SAS and JMP, they recognize a need for integrating more interactive visuals in the model building process. Overall, we believe that such a framework could be applied to a wide range of social media data in which analysts want to locally extract information from social media and use trend values and other metrics as input to their modeling process. We believe that predictive analytics in general can be improved upon by integrating human knowledge into the workflow and can add more transparency to the oftentimes black-box model that encompasses many of the current prediction methods (e.g., SVM).

7 Future work

For future work we want to improve a users understanding of a feature's impact on a model. We also want to develop methods to explore and select features according to multivariate dependencies and feature engineering. Visualization can explain results and reveal complex dependencies. To find such dependencies we want to integrate and orchestrate even more data sources, such as news media and other social media sources like bitly and Facebook, as well as weather and seasonal information such as holidays. Moreover we expect dependencies between past and concurrent weekend releases to be highly important. We also want to focus on the machine learning aspect of prediction. As our models makes structure assumptions, for example, the linear regression model only covered linear relationships, we think we can further improve predictions by investigating the domain data more deeply and use these insights to help analysts choose the right algorithms and options.

8 Acknowledgments

This work was supported by the U.S. Department of Homeland Securitys VACCINE Center under Award Number 2009-ST-061-CI0001. We also wish to acknowledge to the cooperative graduate program Digital Media of the University of Stuttgart, the University of Tuebingen, and the Stuttgart Media University (HdM), as well as the German Federal Ministry of Education and Research (BMBF) in context of the VASA project and the Stuttgart Vast Challenge Team 2013.

REFERENCES

[1] IMDb Database Statistics, 2014. http://www.imdb.com/stats.

[2] R. Amar and J. Stasko. A knowledge task-based framework for design and evaluation of information visualizations. In *IEEE Symposium on Information Visualization*, pages 143–150. IEEE, 2004.

[3] S. Baccianella, A. Esuli, and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the International Conference on Language Resources and Evaluation*, 2010.

[4] D. A. Belsley, E. Kuh, and R. E. Welsch. *Regression diagnostics: Identifying influential data and sources of collinearity*, volume 571. John Wiley & Sons, 2005.

[5] W. Berger, H. Piringer, P. Filzmoser, and E. Gröller. Uncertainty-aware exploration of continuous parameter spaces using multivariate prediction. In *Computer Graphics Forum*, volume 30, pages 911–920. Wiley Online Library, 2011.

[6] M. Bostock, V. Ogievetsky, and J. Heer. D3 data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, Dec. 2011.

[7] E. T. Brown, J. Liu, C. E. Brodley, and R. Chang. Dis-function: Learning distance functions interactively. In *IEEE Conference on Visual Analytics Science and Technology*, pages 83–92. IEEE, 2012.

[8] N. Cao, Y.-R. Lin, X. Sun, D. Lazer, S. Liu, and H. Qu. Whisper: Tracing the spatiotemporal process of information diffusion in real time. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2649–2658, 2012.

[9] J. Chae, D. Thom, H. Bosch, Y. Jang, R. Maciejewski, D. S. Ebert, and T. Ertl. Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In *IEEE Conference on Visual Analytics Science and Technology*, pages 143–152. IEEE, 2012.

[10] J. Choo, H. Lee, J. Kihm, and H. Park. ivisclassifier: An interactive visual analytics system for classification based on supervised dimension reduction. In *IEEE Symposium on Visual Analytics Science and Technology*, pages 27–34. IEEE, 2010.

[11] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[12] A. Culotta. Towards detecting influenza epidemics by analyzing twitter messages. In *Proceedings of the First Workshop on Social Media Analytics*, pages 115–122, New York, NY, USA, 2010. ACM.

[13] W. Dou, L. Yu, X. Wang, Z. Ma, and W. Ribarsky. Hierarchicaltopics: Visually exploring large text collections using topic hierarchies. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2002–2011, 2013.

[14] K. Ericsson and H. Simon. *Protocol Analysis: Verbal Reports as Data*. MIT Press, Cambridge, MA, 1993.

[15] G. Eysenbach. Can tweets predict citations? Metrics of social impact based on Twitter and correlation with traditional metrics of scientific impact. *Journal of medical Internet research*, 13(4), 2011.

[16] S. Geisser. *Predictive inference*. Chapman & Hall, New York, 1993.

[17] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA Data Mining Software: An Update. *SIGKDD Explorations Newsletter*, 11(1):10–18, Nov. 2009.

[18] M. Hao, C. Rohrdantz, H. Janetzko, U. Dayal, D. A. Keim, L. Haug, and M.-C. Hsu. Visual sentiment analysis on twitter data streams. In *IEEE Conference on Visual Analytics Science and Technology*, pages 277–278. IEEE, 2011.

[19] M. Harrower and C. A. Brewer. Colorbrewer. org: An online tool for selecting colour schemes for maps. *The Cartographic Journal*, 40(1):27–37, 2003.

[20] S. Haykin. *Neural networks: A comprehensive foundation*. Prentice Hall PTR, 1994.

[21] A. Inselberg. *Parallel Coordinates: Visual Multidimensional Geometry and Its Applications*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2009.

[22] Jersey Team. Jersey: Restful web services in java. `https://jersey.java.net/`, 2014.

[23] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 1137–1143. Morgan Kaufmann, 1995.

[24] R. Krüger, H. Bosch, D. Thom, E. Püttmann, Q. Han, S. Koch, F. Heimerl, and T. Ertl. Prolix - visual prediction analysis for box office success. In *IEEE Conference on Visual Analytics Science and Technology*, 2013.

[25] D. Lazer, R. Kennedy, G. King, and A. Vespignani. The parable of google flu: Traps in big data analysis. *Science*, 343(6176):1203–1205, 2014.

[26] H. Liu and L. Yu. Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering*, 17(4):491–502, April 2005.

[27] Y. Lu, F. Wang, and R. Maciejewski. VAST 2013 Mini-Challenge 1: Box Office VAST-Team VADER. In *IEEE Conference on Visual Analytics Science and Technology*, 2013.

[28] A. M. MacEachren, A. Jaiswal, A. C. Robinson, S. Pezanowski, A. Savelyev, P. Mitra, X. Zhang, and J. Blanford. Senseplace2: Geotwitter analytics support for situational awareness. In *IEEE Conference on Visual Analytics Science and Technology*, pages 181–190. IEEE, 2011.

[29] T. May, A. Bannach, J. Davey, T. Ruppert, and J. Kohlhammer. Guiding feature subset selection with an interactive visualization. In *IEEE Conference on Visual Analytics Science and Technology*, pages 111–120. IEEE, 2011.

[30] D. C. Montgomery, E. A. Peck, and G. G. Vining. *Introduction to Linear Regression Analysis*. Wiley, 2012.

[31] F. Morstatter, S. Kumar, H. Liu, and R. Maciejewski. Understanding twitter data with tweetxplorer. In *Proceedings of the ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1482–1485. ACM, 2013.

[32] T. Muhlbacher and H. Piringer. A partition-based framework for building and validating regression models. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):1962–1971, 2013.

[33] V. D. Nguyen, B. Varghese, and A. Barker. The royal birth of 2013: Analysing and visualising public sentiment in the UK using Twitter. In *IEEE International Conference on Big Data*, pages 46–54. IEEE, 2013.

[34] C. Okoli and S. D. Pawlowski. The delphi method as a research tool: an example, design considerations and applications. *Information & Management*, 42(1):15–29, 2004.

[35] H. Piringer, W. Berger, and H. Hauser. Quantifying and comparing features in high-dimensional datasets. In *12th International Conference on Information Visualisation*, pages 240–245. IEEE, 2008.

[36] S. Publishing et al. *JMP 10 Modeling and Multivariate Methods*. SAS Institute, 2012.

[37] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2014.

[38] G. Rowe and G. Wright. The delphi technique as a forecasting tool: issues and analysis. *International journal of forecasting*, 15(4):353–375, 1999.

[39] SAS Institute Inc. *SAS/STAT Software, Version 9.3*. Cary, NC, 2011.

[40] J. Seo and B. Shneiderman. A rank-by-feature framework for unsupervised multidimensional data exploration using low dimensional projections. In *IEEE Symposium on Information Visualization*, pages 65–72. IEEE, 2004.

[41] J. S. Simonoff and I. R. Sparrow. Predicting movie grosses: Winners and losers, blockbusters and sleepers. *Chance*, 13(3):15–24, 2000.

[42] D. Thom, H. Bosch, S. Koch, M. Worner, and T. Ertl. Spatiotemporal anomaly detection through visual analysis of geolocated twitter messages. In *IEEE Pacific Visualization Symposium (PacificVis)*, pages 41–48. IEEE, 2012.

[43] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe. Predicting elections with twitter: What 140 characters reveal about political sentiment. *International AAAI Conference on Weblogs and Social Media*, 10:178–185, 2010.

[44] J. Ulbts. JMDB: Java Movie Database. `http://www.jmdb.de/`, 2014.

[45] C. Wang, Z. Xiao, Y. Liu, Y. Xu, A. Zhou, and K. Zhang. Sentiview: Sentiment analysis and visualization for internet popular topics. *IEEE Transactions on Human-Machine Systems*, 43(6):620–630, 2013.

[46] P. Xu, Y. Wu, E. Wei, T.-Q. Peng, S. Liu, J. J. Zhu, and H. Qu. Visual analysis of topic competition on social media. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2012–2021, 2013.

[47] D. Zeng, H. Chen, R. Lusch, and S.-H. Li. Social media analytics and intelligence. *Intelligent Systems, IEEE*, 25(6):13–16, 2010.

[48] X. Zhang, H. Fuehres, and P. A. Gloor. Predicting stock market indicators through twitter "I hope it is not as bad as I fear". *Procedia-Social and Behavioral Sciences*, 26:55–62, 2011.

[49] Z. Zhao, L. Wang, and H. Liu. Efficient spectral feature selection with minimum redundancy. In *AAAI Conference on Artificial Intelligence*, 2010.

Proactive Spatiotemporal Resource Allocation and Predictive Visual Analytics for Community Policing and Law Enforcement

Abish Malik, Ross Maciejewski, *Member, IEEE*, Sherry Towers, Sean McCullough, and David S. Ebert, *Fellow, IEEE*

Abstract— In this paper, we present a visual analytics approach that provides decision makers with a proactive and predictive environment in order to assist them in making effective resource allocation and deployment decisions. The challenges involved with such predictive analytics processes include end-users' understanding, and the application of the underlying statistical algorithms at the right spatiotemporal granularity levels so that good prediction estimates can be established. In our approach, we provide analysts with a suite of natural scale templates and methods that enable them to focus and drill down to appropriate geospatial and temporal resolution levels. Our forecasting technique is based on the Seasonal Trend decomposition based on Loess (STL) method, which we apply in a spatiotemporal visual analytics context to provide analysts with predicted levels of future activity. We also present a novel kernel density estimation technique we have developed, in which the prediction process is influenced by the spatial correlation of recent incidents at nearby locations. We demonstrate our techniques by applying our methodology to Criminal, Traffic and Civil (CTC) incident datasets.

Index Terms—Visual Analytics, Natural Scales, Seasonal Trend decomposition based on Loess (STL), Law Enforcement

1 Introduction

The increasing availability of digital data provides both opportunities and challenges. The potential of utilizing these data for increasing effectiveness and efficiency of operations and decision making is vast. Harnessing this data with effective tools can transform decision making from reactive to proactive and predictive. However, the volume, variety, and velocity of these data can actually decrease the effectiveness of analysts and decision makers by creating cognitive overload and paralysis by analysis, especially in fast-paced decision making environments.

Many researchers in data visualization and visual analytics [37] have proposed interactive visual analytical techniques to aid analysts in these tasks. Unfortunately, most work in this area has required these casual experts (experts in domains, but not necessarily statistics experts) to carefully choose appropriate parameters from a vast parameter space, select the proper resolution over which to perform their analysis, apply appropriate statistical or machine learning analysis techniques, and/or understand advanced statistical significance testing, while accounting for the different uncertainties in the data and processes.

Moreover, the casual experts are required to adapt their decision making process to the statistical analysis space where they need to choose the appropriate time and space scales that give them meaningful analytical and predictive results. They need to understand the role that data sparsity, different distribution characteristics, data variable co-dependencies, and data variance play in the accuracy and reliability of the analytical and prediction results. In moving to this proactive and predictive environment, scale issues become even more important. Not only does the choice of appropriate scales help guide the users' perception and interpretation of the data attributes, it also facilitates gaining new insight into the dynamics of the analytical tasks [42] and the validity of the analytical product: a spatial resolution level that is too fine may lead to zero data input values with no predictive statistical value; whereas, a scale that is too coarse can overgeneralize the data and introduce variation and noise, reducing the value and specificity of the results. Therefore, it becomes critical for forecasting and analysis to choose statistically meaningful resolution and aggregation scales. Utilizing basic principles from scaling theory [42], and Norman's naturalness and appropriateness principles [26], we can both balance and harness these cognitively meaningful natural human-centered domain scales with meaningful statistical scales.

Therefore, in this paper, we present a visual analytics approach that provides casual experts with a proactive and predictive environment that enables them to utilize their domain expertise while exploring their problem and making decisions and predictions at natural problem scales to increase their effectiveness and efficiency in planning, resource allocation, and deployment. Our visual analytics framework [21, 22] provides interactive exploration of multisource, multivariate spatiotemporal datasets using linked views. The system enables the exploration of historic datasets and examination of trends, behaviors and interactions between the different spatiotemporal data elements. The focus of this paper, however, is to provide a proactive decision making environment where historic datasets are utilized at natural geospatial and temporal scales in order to guide future decisions and resource allocation strategies.

In our predictive visual analytics process, we allow users to interactively select and refine the data categories over which to perform their analyses, explore and apply meaningful geospatial (Sections 4.1-4.3) and temporal (Section 4.4) scales and aggregations, apply the forecasting process over geospace (Section 5), and visualize the forecasting results over their chosen geospatial domain. We utilize a Seasonal Trend decomposition based on Loess (STL) [9] approach (Section 3) that utilizes patterns of historical data and apply it in the geospatial domain to predict future geospatial incidence levels. Moreover, this approach provides domain-driven refinement of analysis and exploration to areas and time of significance (e.g., high crime areas or times).

The contributions of our work include these novel natural spatial and temporal analytical techniques, as well as a novel Dynamic Covariance Kernel Density Estimation method (DCKDE) (Section 4.2.2). These contributions can be applied to a variety of spatiotemporal datasets including distribution and logistics, public safety, public health, and law enforcement. We will utilize data from Criminal, Traffic, and Civil (CTC) incident law enforcement datasets in the examples throughout this paper. However, it should be noted that our technique is versatile and can be adapted for other relevant spatiotemporal datasets that exhibit seasonality.

- *Abish Malik, Sean McCullough and David S. Ebert are with Purdue University. E-mail: amalik|mccullo0|ebertd@purdue.edu.*
- *Ross Maciejewski and Sherry Towers are with the Arizona State University. E-mail: rmacieje|smtowers@asu.edu.*

Manuscript received 31 March 2013; accepted 1 August 2013; posted online 13 October 2013; mailed on 4 October 2013.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

2 Related Work

In recent years, there has been much work done in utilizing historic datasets for informing future actions and decisions of decision makers. Below, we discuss previous work in the field of visual analytics, and, since our chosen example domain and implementation is focused on crime data, we also explore previous work in criminology to provide a breadth of the related research areas.

2.1 Predictive Visual Analytics

There have been several visual analytics systems developed in recent years that support data analysis and exploration processes, and provide extensive data modeling and hypothesis generation tools (e.g., [28, 35]). More recently however, researchers have also started progressing toward creating visual analytics systems that incorporate predictive analytics in them. For example, Wong et al. [43] provide a visual interface and an environment that brings together research from several different domains to predict and assess the impact of climate change on U.S. power-grids. Muhlbacher and Piringer [25] provide a visual analytics framework for building regression models. Monroe et al. [23] utilize user-driven data visualizations that enable researchers to gain insights into large healthcare datasets.

Yue et al. [45] created an artificial intelligence based tool that leverages interactive visualization techniques to leverage data in a predictive analytics processes. Their time series modeling technique includes the use of the Box-Jenkins procedure [27]. Other time series modeling techniques extensively used include the ARMA (Auto Regressive Moving Average) [1] and ARIMA (Auto Regressive Integrated Moving Average) models. A summary of some other methods that involve geospatial modeling can be found in [11, 12]. Maciejewski et al. [20] utilize the seasonal trend decomposition by loess smoothing for generating temporal predictions for modeling spatiotemporal healthcare events. They also use the kernel density estimation technique for creating probability distributions of patient locations for use in healthcare data. Our work builds on these ideas where we utilize historic datasets to provide spatiotemporal forecasts into the future. The focus of our work is to explore the issues of geospatial and temporal scales so that casual experts can adapt their decision making process to the statistical analysis space. As such, we apply a user assisted data analysis approach to drive future decisions that helps prevent decision makers from getting over-burdened, while, at the same time, maximizes the utilization of their domain knowledge and perceptual capabilities.

2.2 Crime Hotspot Policing and Intervention

In recent years, there has been much research done that suggest the benefits of hot spot policing in preventing crime and disorder at these crime hotspots (e.g., [2, 3, 4]). Weisburd et al. [41] examine the effect and impact of crime hot spots policing and their findings suggest little negative effects and backlash among the residents of targeted areas of such policing efforts. Sherman [30] also explores the effects of police crackdowns (sudden increase in police presence in specific regions) among several case studies. He notes that while most of the crackdowns appeared to demonstrate initial deterrent effects, the effects decayed after short periods of time. Our work also enables law enforcement decision makers to identify and target crime hotspots by forecasting high probability crime regions based on historic spatiotemporal trends. Our work also factors in the temporal variations within the signals and, as such, provides dynamic hotspot locations for each predicted day.

Goldkamp and V$\hat{i}$lcic$\tilde{a}$ [15] provide insights into unanticipated negative effects of place-oriented enforcement intervention schemes on other societal aspects. They explored an intensive targeted enforcement strategy that was focused on drug crime and its related community effects and examined the overall side effects on the society. Sherman et al. [31] examine and provide an overview of the different aspects of predatory criminal activity at different spatial granularities and how these factors correlate with different aspects of the society. Bruin et al. [7] provide a toolkit that extracts the different factors from police datasets and creates digital profiles for all offenders. The tool then clusters the individuals against the created profiles by using a distance matrix that is built around different attributes (e.g., crime frequency, criminal history of the offenders).

2.3 Predictive Policing

There has been much work done in criminology to study criminal behaviors in order to develop models that predict various offense incidence levels at different spatial aggregation levels. Brown and Oxford [6] study methods that pertain to predicting the number of breaking and enterings in sub-cities and correlate breaking and enterings with different factors including unemployment rates, alcohol sales and previous incidents of crime. Yu et al. [44] also develop a crime forecasting model by employing different data mining classification techniques. They employ several classification techniques including Nearest Neighbor, Decision Tree and Support Vector Machines. Their experiments are run on two different data grid sizes, the 24-by-20 (approx. one-half mile square) and the 41-by-40 square grid cells (approx. one-quarter mile square). They note that the 24-by-20 grids consistently gave them better results than the 41-by-40 grids, which they attribute to the lack of sufficient information at the coarser resolution. Our technique also allows analysts to conduct their predictive forecasting at different spatial resolutions (e.g., over uniform spatial grids and natural underlying spatial boundaries) and temporal granularity levels (e.g., by day, week, month). Furthermore, our system also allows users to create spatial and temporal templates for use in the prediction process.

Monthly and seasonal cycles and periodic properties of crime are well known among criminologists [17]. Felson and Poulson [14] factor in the time of the day variation in the analysis of crime and provide summary indicators that summarize the hour-of-day variations. They provide guidelines for breaking the day into quartiles based on the median hour of crime. We use their guidelines in our work and provide default data driven time-of-day templates over which to forecast crime. We also utilize these techniques and incorporate the seasonality and periodicity properties of crime in order to provide spatiotemporal forecasts of future crime incidence levels.

3 Time Series Prediction using Seasonal-Trend Decomposition based on Loess (STL)

In order to model time series data, we employ the seasonal-trend decomposition technique based on a locally weighted regression (loess) methodology (STL), where a time series signal is considered to consist of the sum of multiple components of variation. To accomplish this, we first utilize the STL method [9, 16] to desynthesize the time series signal into its various components. An analysis of the underlying time series signal Y for CTC data reveals that a square root power transform stabilizes the variability and yields a more Normal distribution of time series residuals, which is a requirement to appropriately model the time series using STL. We consider the time series signal $\sqrt{Y}$ to consist of the sum of its individual components given by $\sqrt{Y_v} = T_v + S_v + D_v + R_v$, where, for the v-th time step, T_v is the inter-annual component, S_v is the yearly-seasonal component, D_v is the day-of-the-week effect, and R_v is the remainder variation component.

To predict using the STL method, we apply the methodology described in [20], where the fitted values $\hat{Y} = (\hat{y}_1, ..., \hat{y}_n)$ generated using the loess operator in the STL decomposition step are considered to be a linear transformation of the input time series $Y = (y_1, ..., y_n)$. This is given by $\hat{y}_i = \sum_{i=1}^{n} h_{ij} y_j \Rightarrow \hat{Y} = HY$, where H is the operator matrix whose (i, j)-th diagonal elements are given by $h_{i,j}$. In order to predict ahead by n days, we append the operator matrix H obtained from predicting ahead within each linear filter in the STL process with n new rows, and use this to obtain the predicted value. The predicted value for day $n+1$ is thereby given by $\hat{y}_{n+1} = \sum_{i=1}^{n} H_{n+1,i}\, Y_i$.

We use this concept of time series modeling and prediction and extend it into the spatiotemporal domain (see Section 5 for details). We further factor in for the sparcity of data in certain geographical regions, and devise strategies to alleviate problems resulting in prediction in these sparse regions (Section 4).

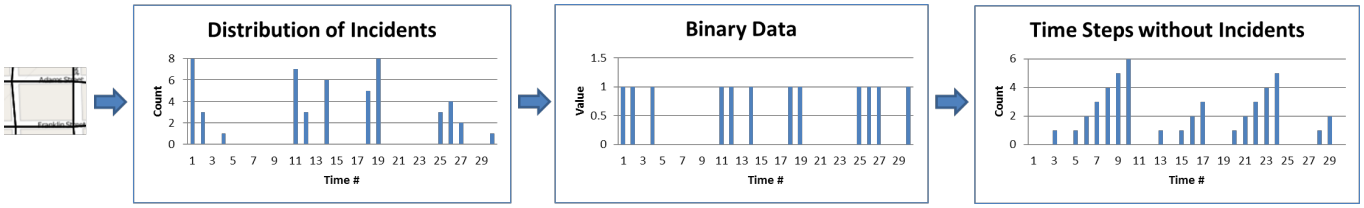


Fig. 1. Our geospatial natural scale template signal generation process. For each geospatial sub-division, the system generates a time series of the number of incidents, converts it into a binary signal, and processes the binary signal to generate the signal used to form the geospatial template.

4 Natural Scale Templates

In order to assist with the analysis process, we provide decision makers with natural scale templates that enable them to focus on appropriate geospatial and temporal resolution levels. These templates enable users to analyze their data at appropriate spatiotemporal granularity levels that help align the scale and frame of reference of the data analysis process with that of the decision making process. These templates also assist users in alleviating the impedance mismatch between data size/complexity and the decision makers' ability to understand and interact with data [29]. We support the creation of both geospatial and temporal templates in our system that facilitate the decision making process. A combination of the generated geospatial and temporal templates provide analysts with an appropriate starting point in the analysis process; thereby, eliminating the need to examine and analyze the entire spatiotemporal parameter space and reducing it to more manageable, appropriate scale levels. To be effective, the design of these scale templates must follow the appropriateness, naturalness, and matching cognitive principles [26]. As Wilkinson and Stevenson both point out [36, 40, 42], simple scaling theory techniques are not sufficient (e.g., axometric scaling theory), but provide useful guidance to primitive scales of reference. The combinations of these design principles and the guidance from these statistical scale papers, provide the motivation and basis for our natural scale templates described below.

4.1 Geospatial Templates

An underlying assumption with using STL to decompose time series is that the data are Normally distributed. The model predictions can get severely biased if this assumption is violated or if data are sparse. To remedy this, we provide methods that help guide users in creating geospatial scales that allow them to drill down to higher incidence regions that may provide better prediction estimates.

4.1.1 Geospatial Natural Scale Templates based on Spatiotemporal Incident Distribution

Our system allows users to narrow down the geographic space for the scope of analysis to regions with higher incidence counts and higher statistical significance for user-selected incident types. Our geospatial natural scale template methodology is shown in Figure 1. In order to generate geospatial templates, the system first fragments the geographic space into either uniform rectangular grids [6] or man-made spatial demarcations (e.g., census blocks). Then, for each subregion, the system generates a time series of the number of incidents that occurred within the subregion over time (e.g., by day, week, month). This signal is further cached for use later in the forecasting process. Next, we convert this time series signal into a binary signal across time, where a 1 represents that an incident occurred on a particular day and a 0 that no incident occurred. We then count the number of 0's between the 1's and progressively sum the number of 0's, outputting the result as another time series signal. As such, this signal is a representation of the number of time steps over which no incidents occurred for the given subregion.

This new time series signal is now utilized in the STL forecasting method (Section 3) and a predicted value is computed for the next day. It should be noted that the resulting time series for regions of lower incidence counts will not be sparse, and consequently, will generate higher predicted values. This process is repeated for all geospatial subregions and a unified picture is obtained for the next day. Finally, we filter out the regions with higher predicted values (low activity) by thresholding for the maximum value. The resulting filtered region forms the initial geospatial template. An example of a created geospatial template using this technique is shown in Figure 4 (Left).

4.1.2 User Refinement of Geospatial Template using Domain Knowledge

The geospatial template provides regions with relatively higher incident rates. The system further allows users to use their domain knowledge and interactively refine these template regions into sub-divisions. For example, users may choose to sub-divide the formed template regions by natural or man-made boundaries (e.g., state roads, rivers, police beats), or by underlying features (e.g., known drug hotspots). The system also allows users to explore the predicted future counts of the created sub-regions by generating an incidence count vs. time signal for each disjoint region and applying our forecasting methodology (Section 3) to find a predicted value for the next day. The results are then shown as a choropleth map to users (e.g., Figure 4 (Right)). These macro-level prediction estimates further assist decision makers in formulating high-level resource allocation strategies.

4.2 Kernel Density Estimation

One of the challenges with using the spatial distribution of incidents in a geospatial predictive analytics process is that it can exacerbate the problem of generating signals with low or no data values. To further refine our prediction model in geospace, we utilize a Kernel Density Estimation (KDE) technique to spread the probability of the occurrence of incidents to its neighboring regions. The rationale behind this is that criminology research has shown evidence that occurrence of certain types of crimes (e.g., residential burglary) at a particular region puts neighboring regions at an elevated risk [13, 18, 32].

Furthermore, crime also tends to be clustered in certain neighborhoods, and the probability of a crime occurring at a particular location can be highly correlated with the number of recent crimes at nearby locations. We incorporate this concept in a novel kernel density estimation method described in Section 4.2.2, where the kernel value at a given location depends on the locations of its k-nearest incidents. In addition, kernel density estimation methods take into account that crimes in low-crime or sparsely populated areas have low incidence, but non-zero probability. We utilize two interchangeable density estimation techniques in our implementation.

4.2.1 Kernel Scale based on Distance to the k-th Nearest Neighbor

To account for regions with variable data counts, we utilize a kernel density estimation technique and use a dynamic kernel bandwidth [33]. We scale the parameter of estimation by the distance from the point x to its kth nearest neighbor X_i. This is shown in Equation 1.

$$\hat{f}(x) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{max(h,d_{i,k})}K\left(\frac{x-X_i}{max(h,d_{i,k})}\right) \quad (1)$$

Here, N is the total number of samples, $d_{i,k}$ the distance from the i-th sample to the k-th nearest neighbor and h is the minimum allowed kernel width. We use the Epanechnikov kernel [33] to reduce calculation time, which is given by $K(\mathbf{u}) = \frac{3}{4}(1-\mathbf{u}^2)1_{(||u||\leq 1)}$. Here, the

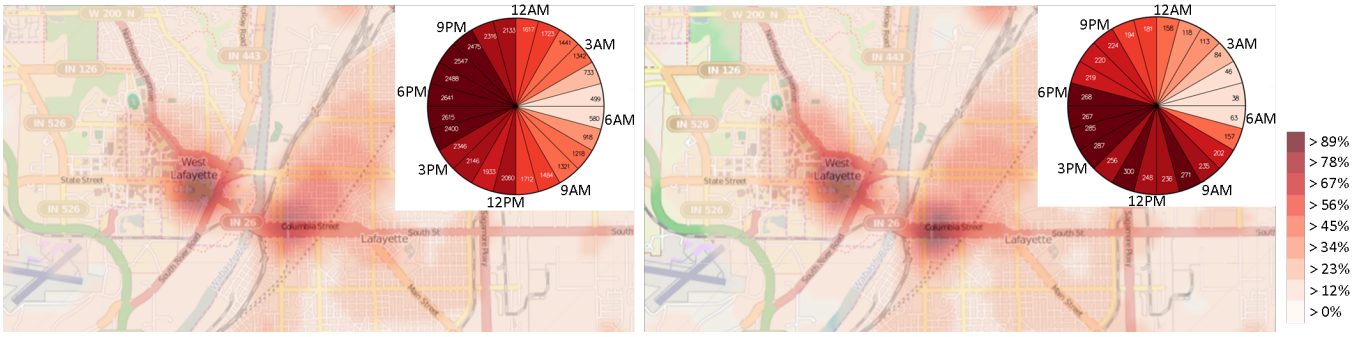


Fig. 2. Spatiotemporal distribution of historical CTC incidents for Tippecanoe County for (Left) 3/11/2012 through 3/10/2014, and (Right) for all Tuesdays in March in the past 10 years.

function $1_{(||u||\leq 1)}$ evaluates to 1 if the inequality is true and to 0 otherwise. In cases where the distance from the i-th sample to the k-th nearest neighbor is 0 (e.g., multiple calls from the same address), we force the variable kernel estimation to a minimum fixed bandwidth h. Making the kernel width placed at the point X_i proportional to $d_{i,k}$ gives regions with sparse data a flatter kernel, and vice-versa.

4.2.2 Dynamic Covariance Kernel Density Estimation Technique (DCKDE)

The kernel in the previous method is based on the distance from an incident location to its k-th nearest neighbor, which provides a flatter kernel for sparse regions. In a new kernel method, we use the information from all k-nearest neighbors to calculate the width of the kernel (rather than the most distant neighbor), thus reducing stochastic variation on the width of the kernel. As such, we fragment the geospatial region into rectangular grids and then utilize a Gaussian kernel at every grid node that is based on the covariance matrix of the location of the center of each node $\mathbf{X} = \{x, y\}$ and its k-nearest neighbor incidents [39]. Therefore, the kernel value is influenced by the k-nearest neighbors and provides a wider kernel in sparsely populated regions that enables the model prediction to be small but non-zero and also takes into account correlations between latitude and longitude; thus, improving the accuracy of the estimates. The value stored at each node location is given by $\delta(\mathbf{X}) = \frac{1}{2\pi|V|} e^{-\frac{1}{2}(\mathbf{X}-\boldsymbol{\mu})^T V^{-1}(\mathbf{X}-\boldsymbol{\mu})}$, where $\boldsymbol{\mu} = \{\mu_x, \mu_y\}$ is the mean along the x and y directions of the k nearest neighbors and their covariance matrix V is defined as:

$$V = \begin{bmatrix} \sigma_x^2 & cov_{x,y} \\ cov_{x,y} & \sigma_y^2 \end{bmatrix} \tag{2}$$

Here, σ_x^2 and σ_y^2 is the variance along the x and y dimension respectively, and $cov_{x,y} = \sum_{i=1}^{k} \frac{(x_i-\mu_x)(y_i-\mu_y)}{k-1}$ is the sample covariance between x and y.

4.3 Neighbors with Similar Spatio-Demographics

For regions that generate a signal of lower statistical significance for the user selected categories, we provide the option to explore data in similar neighborhoods. For each census block, we utilize spatio-demographic census data to find those census blocks that exhibit similar spatial demographics. The rationale behind finding similar neighborhoods lies in the fact that regions with similar demographics tend to exhibit similar trends for certain types of crime [24, 34].

The process of finding similar census blocks for a given census block X includes computing the similarity distance from X to all neighboring census blocks that lie within a d mile radius from the centroid of X. The d mile radius constraint is imposed to factor in for Tobler's first law of geography [38] that suggests that near regions are more related to one another than distant regions. We use $d = 3.0$ miles in our implementation [8]. As such, the similarity distance between two census blocks A and B given k census data variables is given by $S_{A,B} = \sqrt{\sum_{i=1}^{k}(A(V_i) - B(V_i))^2}$, where $A(V_i)$ and $B(V_i)$ are the corresponding census data variable values (e.g., race, income, and age demographic data) for census blocks A and B respectively. Finally, the top N census blocks with the smallest similarity distance values are chosen as the similar census blocks for the given census block X. We use $N = 5$ as a default value in our implementation, but provide users with options to change this value on demand. We note that our future work includes extending this concept of finding similar neighborhoods to determining similar data categories for predictive purposes.

The system now provides users with the ability to generate *similar neighborhood* prediction maps where the prediction for a given census block X depends on the historic time series data of its N similar census blocks in addition to the past data of the census block X itself. Here, the input time series for the census block X used in the prediction algorithm is the per time step average of the N similar census block signals combined with the original signal from census block X. The resulting prediction maps incorporates the influence of incidence rates in neighborhoods that share similar spatio-demographic data.

4.4 Temporal Natural Scale Templates

As noted previously in Section 2.3, crime trends exhibit not only monthly and seasonal trends, but also shows day-of-the-week and hour-of-day variations. The prediction maps produced by the methods described so far provide prediction estimates over 24-hour periods. This information, albeit valuable to the law enforcement community in developing resource allocation strategies for their precincts, provides little detail of the 24-hour distribution of crime. In this section, we describe our approach to assist users in creating temporal scales.

4.4.1 Interactive Clock Display

Figure 2 (Top-Right) shows our interactive clock view that enables a radial display of temporal hourly data. The clock view provides a way for users to filter the data by the hour by interactively clicking on the desired hours, thereby filtering down the data for use in the prediction process. Users may use the clock view display to obtain a visual summary of the hourly distribution of the incidents and consequently make informed decisions on creating temporal templates over which good prediction estimates may be established.

4.4.2 Factoring in for Monthly and Day-of-the-Week Variations

In addition to utilizing the seasonal trend decomposition technique described in Section 3 to decompose the time series signals into its various components, we also utilize a direct approach where we allow users to create their own custom monthly and/or daily templates. Certain crimes tend to peak on certain days of the week (e.g., alcohol related violations tend to be higher over the weekend), whereas other crimes tend to be lower on other days (e.g., reported burglaries drop over the weekend). As such, we factor for these effects directly in the system and allow users to filter data specifically by month and/or by day-of-the-week. This further assists decision makers in developing and refining their resource allocation strategies.

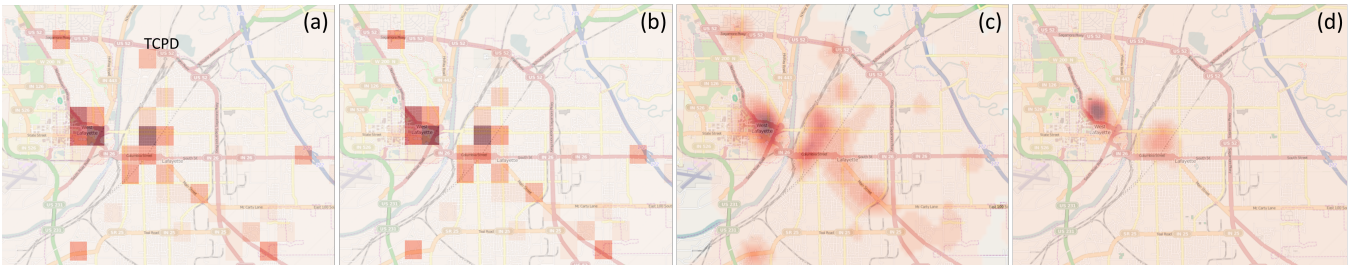


Fig. 3. Geospatial prediction results for 3/11/2014 for Tippecanoe County obtained using our STL forecasting methodology. (a) Predicted choropleth map for rectangular grids of dimension 64×64 using incidence count time series by day. (b) Refined predicted map after removing TCPD location from (a). (c) Predicted map using KDE based on the distance to the k-th nearest neighbor approach (Section 4.2.1). (d) Forecast map using DCKDE method (Section 4.2.2).

4.4.3 Refinement using Summary Indicators

We extend the method described in [14] to further assist users with refining and choosing appropriate hourly templates in the prediction process. In this method, the system computes the *median minute* of CTC incident for the selected 24-hour binning period that provides information about when exactly half of the incidents for the selected date range and offense types have occurred. Next, to get an indication of the dispersion of crime within the 24-hour period, the system computes the *first quartile minute* and *third quartile minute* for the selected data, which are the median times of the first and second halves of the 24-hour period from the median minute respectively. Finally, as temporal data can be inaccurate with many incidents that have missing time stamps, we provide users with an accuracy indicator to show the percentage of cases with valid time stamps. These summary indicators, along with the temporal templates described above, enable users to further refine their selected temporal templates for use in the prediction process. Example scenarios where these indicators are used are provided in Section 6.

5 Geospatial Prediction

The described visual analytics process involves a domain expert selecting appropriate data parameters, applying desired data filters and generating spatial and temporal natural scale templates using the methods described in Section 4. Next, the system incorporates the STL forecasting method (Section 3) and extends it to the geospatial domain to provide prediction estimates for the next N time steps (e.g., days, weeks, months). We now list the steps involved in our geospatial prediction methodology:

1. **Dividing geospace into sub-regions**: The first step in our methodology, just like in Section 4.1.1, involves subdividing geospace into either uniform rectangular grids of user specified resolutions or man-made geospatial boundaries.
2. **Generating the time series signal**: The system then extracts a time series signal for each sub-division. We allow two types of signals to be extracted for each sub-division: (a) incidence count vs. time step, and (b) kernel value vs. time step. Note that the signal generated in (a) is the same as that produced in Section 4.1.1 (Figure 1 (Distribution of Incidents)). The kernel values used in (b) are generated using any one of the methods described in Section 4.2.
3. **Forecasting**: The time series signal generated for each spatial unit is then fed through the STL process described in Section 3 where a forecast is generated for the next N time steps (e.g., days, weeks). This process is repeated for all region sub-divisions and prediction maps are finally obtained for the next N time steps.
4. **Visualizing results**: Finally, the results of our forecasting method are provided to the user either in the form of a choropleth map or a heatmap.

When users choose to fragment the geospace into uniform rectangular grids, we provide them with the ability to select the resolution level, or, in other words, the grid size of each grid. An incidence count vs. time step signal is then generated for each sub-region. It is important to note here that a grid resolution that is too fine may result in a zero count vs. time step signal that has no predictive statistical value. On the other hand, a grid resolution that is too coarse may introduce variance and noise in the input signal, thereby over-generalizing the data. An evaluation of our forecasting approach (Section 7) indicates that an average input size of 10 samples per time step provide enough samples for which our method behaves within the constraints and assumptions of our STL forecasting approach. We utilize this metric in our system in order to determine the applicability of our forecasting method for a particular sub-region.

Figure 3 shows a series of examples that demonstrate our geospatial prediction results using the methods described in this section. Here, the user has selected all CTC incidents for Tippecanoe County, IN, and is using 10 years' worth of historical data (3/11/2004 through 3/10/2014) to generate forecast maps for the next day (i.e., for 3/11/2014). Figure 3 (a) shows the prediction results when Tippecanoe County, IN is fragmented into rectangular grids of dimension 64×64. The input data for each sub-region consists of daily incidence count data over the last 10 years. This method, unlike the KDE methods, does not spread the probability to surrounding neighborhood regions when an incident occurs at a particular place. As a result, this method treats each region independently, and can be used when there are no correlations between geospatial regions (e.g., commercial vs. residential neighborhoods). This method can also be useful in detecting anomalous regions and regions of high predicted levels of activity. For example, the user notices something peculiar from the results in Figure 3 (a): a predicted hotspot occurs prominently over the Sheriff's office and county jail location (labeled as TCPD in Figure 3 (a)). This occurs because the default geospatial location of many incidents are logged in as the county jail, especially when arrests are associated with cases. To remedy for this, the user can refine the underlying geospatial template (Section 4.1.2) and dynamically remove this location from the geospatial template. The refined prediction map generated is shown in Figure 3 (b).

Figures 3 (c and d) show the predicted results of using the kernel density estimation based on the distance to the k-nearest neighbor approach (Section 4.2.1) and the DCKDE technique (Section 4.2.2), respectively. The KDE method applied to generate the prediction map in Figure 3 (c) provides a flatter kernel for relatively low-crime regions. As a result, the prediction map provides lower, but non-zero, predictions for these regions. The kernel width computed using this method is based on the distance from a point x to its kth nearest neighbor only. The DCKDE method, on the other hand, assumes that the probability of the occurrence of an incident at a particular location is correlated with the number of recent incidents at nearby locations. Accordingly, this method utilizes information from *all* k-nearest neighbors in calculating the kernel value. Thus, the regions with persistently higher incident concentrations generate focused hotspots when forecasting is performed using the DCKDE method. Finally, it should be noted that each method provides users with different insights into the dynamics of the underlying processes, and users can use their domain knowledge to further refine the results to make informed decisions.

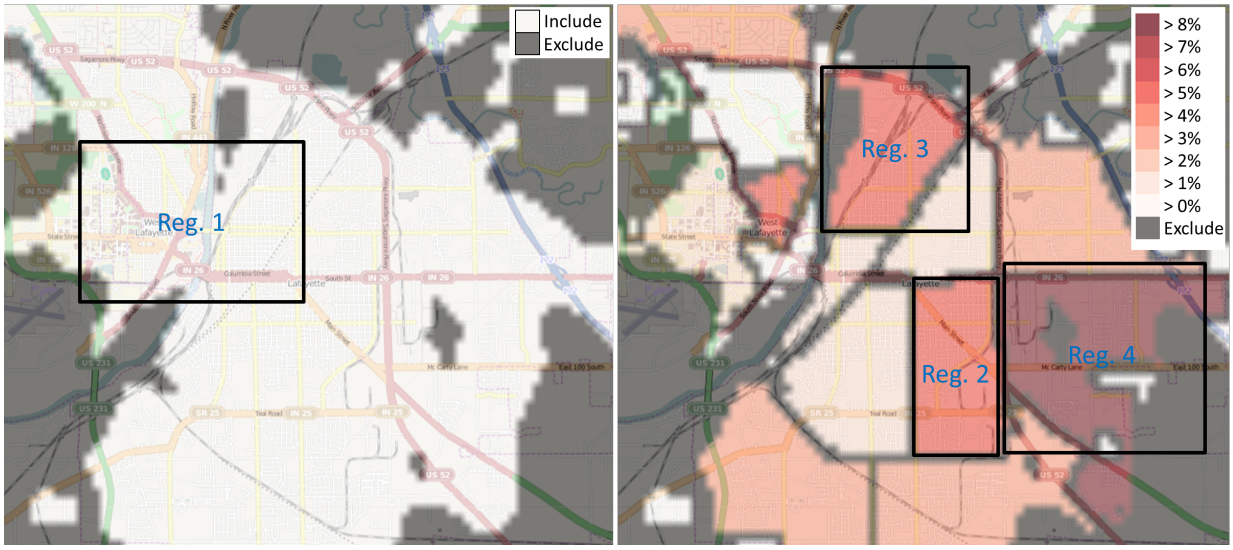


Fig. 4. (Left) Geospatial template generated for Tippecanoe County using 10 years' worth of historical data. (Right) Choropleth map showing the distribution of predicted incidents for 3/11/2014 by police beats for Tippecanoe County. Users may further select regions on the map (e.g., Reg. 1-4) to generate detailed predictions for the selected regions (Figure 5).

6 Case Study: Forecasting Future Criminal, Traffic and Civil (CTC) Incidence Levels

In this section, we demonstrate our work by applying our spatiotemporal natural scale template methodology to forecast for CTC incidence levels in Tippecanoe County, IN, U.S.A. This dataset consists of historical reports and provides several different attributes, including the geographic location, offense type, agency, date, and time of the incident. This dataset contains an average of 31,000 incidents per year for Tippecanoe County, and includes incidence reports for different categories of CTC incidents (e.g., crimes against person, crimes against property, traffic accidents). We use 10 years worth of historical data for this analysis. We provide a workflow when using our system in the analysis process.

Forecasting for all geospatial CTC incidents

Here, we describe a hypothetical scenario in which a law enforcement shift supervisor is using our system to develop resource allocation strategies for Tippecanoe County over the next 24 hour period for Tuesday, March 11, 2014. The supervisor is interested in developing a high-level resource allocation strategy, in particular, by police beats for the next 24 hour period. Law enforcement officers are generally assigned to a particular law beat and patrol their beat during their shift hours when not responding to a call for service. The supervisor is also interested in determining which hotspot locations to focus on for larger police beats. Finally, he also wants to refine the developed resource allocation strategy to factor in for the hourly variation of crime. To develop an appropriate resource allocation strategy, the shift supervisor performs several different analyses that are described in the following subsections. Although our example uses data for all CTC categories as inputs, users may filter their data using any combinations of CTC categories (e.g., crimes against property, person) to further refine their resource allocation strategy.

Overall daily resource allocation

The shift supervisor begins his process by visually exploring the spatiotemporal distribution of historical incidents using our system. When working through the system, the supervisor then visualizes the geospatial and hourly distribution of the incidents that occurred over the past 2 years, as shown in Figure 2 (Left). The supervisor notes several hotspots emerge for the selected period. The locations of these hotspots match with his domain knowledge of the area (e.g., city downtown regions, shopping center locations across town). The static image of the aggregate data, however, does not factor in the inherent spatiotemporal data variations, and basing a resource allocation decision on this image alone would be insufficient. The supervisor is also aware of the fact that police presence can act as a deterrent for certain types of crimes, and, therefore, wants to diversify and maximize police presence in these hotspot areas.

Next, the supervisor wants to factor for monthly and day-of-the-week patterns in his analysis. As such, he visualizes the geospatial and hourly distribution of all CTC incidents that occurred on any Tuesday in the month of March over the past 10 years (Section 4.4.2). The result is shown in Figure 2 (Right). The supervisor notes a slightly different geospatial distribution emerges as a result, with the intensity of hotspots shifting towards the east downtown Lafayette region. In this case, it also becomes apparent that for the 24-hour distribution, 10 AM, 1 PM and 3 PM-6 PM emerge as high activity hours.

Allocating resources by police beats

In order to narrow down the geospace and focus on relevant geographic locations, the supervisor decides to apply our geospatial template generation technique (Section 4.1) with all CTC incidents selected using 10 years' worth of historical data (i.e., from 3/11/2004 through 3/10/2014). The resulting geospace generated is shown in white in Figure 4 (Left). The supervisor notes that the resulting regions correspond to highly populated areas, and exclude areas of infrequent occurrences. Next, the system provides a total predicted number of incidents, N, for March 11, 2014 for the filtered geospatial region. This is done by generating a total incidence count vs. day time series signal using the past 10 years' worth of data and applying the STL forecasting method described in Section 3. Here, N is 59 incidents.

Next, the supervisor is interested in obtaining a high level overview of the distribution of the predicted incidents over geospace, and, in particular, by police patrol routes. As such, the supervisor uses our system and fragments the generated geospatial template using the city law beats shapefile. The resulting geospace is shown in Figure 4 (Right). In order to distribute the total predicted 59 incidents across police beats, the system computes an incidence count vs. day time series signal for each disjoint geospatial region and computes the predicted number of incidents n_i for each region (Section 3). Next, the probability of an incident within each disjoint region is calculated using the formula $p_i = n_i/N * 100$. The results of this operation are then shown to the user as a choropleth map, where each disjoint region is colored according to its value on a sequential color scale [5] (Figure 4 (Right)).

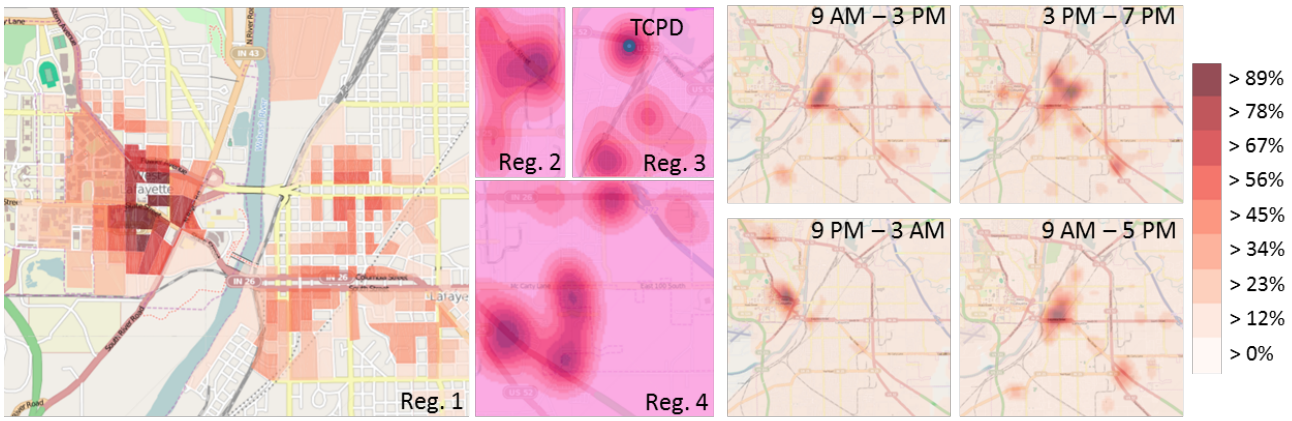


Fig. 5. User refinement of geospatial resource allocation strategy. The user has chosen to visualize predicted hotspots for regions labeled in Figure 4 (Regions 1 through 4), and for Tippecanoe County over hourly temporal templates.

Geospatial resource allocation strategy refinement using domain knowledge

While the high level police beat prediction map (Figure 4 (Right)) suggests putting a heavier emphasis on the eastern police beats of the city, the prediction results in Figure 3 indicate a more localized concentration of incidents at the city downtown locations. The shift supervisor may use these results and allocate higher resources to the eastern police beat of the city (Reg. 4 in Figure 4), and allocate a smaller number of resources, but at more concentrated locations in the downtown (Reg. 1 in Figure 4).

Now, the supervisor is interested in further refining her geospatial resource allocation strategy. First, she turns to the predicted hotspot regions in the city downtown regions (Reg. 1 in Figure 4). She decides to utilize the census blocks spatial boundary information and divides the geospace into census blocks. Next, she uses the method described in Section 5 to create a predicted choropleth map based on census blocks for the region. The result of this operation is shown in Figure 5 (Reg. 1). Here, the supervisor has chosen to use the kernel values obtained from the method described in Section 4.2.1 and spread them across the underlying census blocks for generating these results.

To obtain detailed predictions for the eastern city police beat region (Reg. 4 in Figure 4), the shift supervisor uses a different approach where she draws a region around the selected beat using the mouse and restricts the forecast to the selected region. The result of this operation is shown in Figure 5 (Reg. 4). From domain knowledge, she knows that this area has a high concentration of shopping centers. The hotspots obtained in Figure 5 (Reg. 4) align with these locations. Finally, the supervisor generates similar heatmaps for regions labeled as Reg. 2 and 3 in Figure 4, the results of which are shown in Figures 5 (Reg. 2 and 3), respectively. Note that the county jail location is once again a hotspot in Figure 5 (Reg. 3). With these detailed results in hand, the shift supervisor is able to devise an optimal resource allocation strategy for the next 24 hour period in Tippecanoe County.

Applying temporal templates

Finally, in order to refine her resource allocation strategy to different portions of the day, the shift supervisor chooses to apply the summary indicators method (Section 4.4.3). She finds that the first, median, and third quartile minutes for CTC incidents that occurred in the past 10 years were 9:25 AM, 3:11 PM and 7:28 PM respectively. She also notes that these indicators correspond with the hourly distribution of incidents using the clock view display in Figure 2. Therefore, the supervisor chooses two hourly templates using these summary indicators: (a) 9 AM through 3 PM, and (b) 3 PM through 7 PM. The supervisor also creates two other hourly templates: 9 PM through 3 AM to capture night time activity, and 9 AM through 5 PM to capture working hours of the day. She then uses the kernel density estimation method (Section 4.2.1) and re-generates prediction maps for March 11, 2014. These results are shown in Figure 5. As expected, the supervisor notes the shift in hotspot locations through the 24 hour period, which further enables the refinement of the resource allocation strategy for the different portions of the 24 hour period.

7 Model Evaluation and Validation

In order to evaluate our methodology, we conducted a series of statistical tests to understand the behavior and applicability of our approach in the spatiotemporal domain. Our validation strategy involved testing for the empirical rule of statistics, which describes a characteristic property of a Normal distribution: 95% of the data points are within the range $\pm 1.96\ \sigma$ of μ, where μ and σ are the mean and standard deviation of the distribution, respectively [10]. In order to help alleviate the challenges resulting due to the sparseness of the underlying data, we performed our analyses over a weekly data aggregation level. Our approach involved testing whether the 95% prediction confidence interval bound acquired for the geospatial predictions using our forecasting approach holds when compared against observed data [19]. This confidence bound would be violated if the variance of the observed data is higher (i.e., overdispersed data) or lower (i.e., underdispersed data) than that dictated by the prediction confidence bound. When the 95% prediction bounds are met as expected, and the data conforms to the Normal regime, the applicability of our spatiotemporal STL forecasting method is established.

Building on our STL based time series prediction discussion from Section 3, the variance of the fitted values $\hat{Y} = (\hat{y}_1, ..., \hat{y}_n)$ using the loess operator in the STL decomposition step is given by $Var(\hat{Y}_i) = \hat{\sigma}^2 \sum_{j=1}^{n} H_{ij}^2$ [20]. Here, $\hat{\sigma}^2$ is the variance of the input time series signal Y, and is estimated from the remainder term R_v. Subsequently, the variance for the predicted value $\hat{Y}_{n+1}$ for time step $n+1$ is given by $Var(\hat{Y}_{n+1}) = \hat{\sigma}^2(1 + \sum_{j=1}^{n} H_{n+1,j}^2)$. This provides the 95% prediction interval as $CI_{n+1} = \hat{Y}_{n+1} \pm 1.96\sqrt{Var(\hat{Y}_{n+1})}$.

Next, we performed a series of analyses at varied geospatial and temporal scales, and for different data categories. The geospace was first fragmented into sub-regions (either rectangular grids or using man-made boundaries), and time series signals were generated for each geospatial sub-region. In our analyses, we utilized a sliding time window of size 3 years (i.e., 3×52 weeks) that provided enough samples above the Nyquist frequency for the STL forecasting technique. Forecasting was performed using the methods described in Sections 5 and 7.1. We provide our evaluation methodology and results in the subsequent sub-sections.

7.1 Modified STL forecasting method to factor in for weekly data aggregation

As described earlier in Section 3, a time series signal $\sqrt{Y}$ can be considered to consist of the sum of its inter-annual (T_v), yearly-

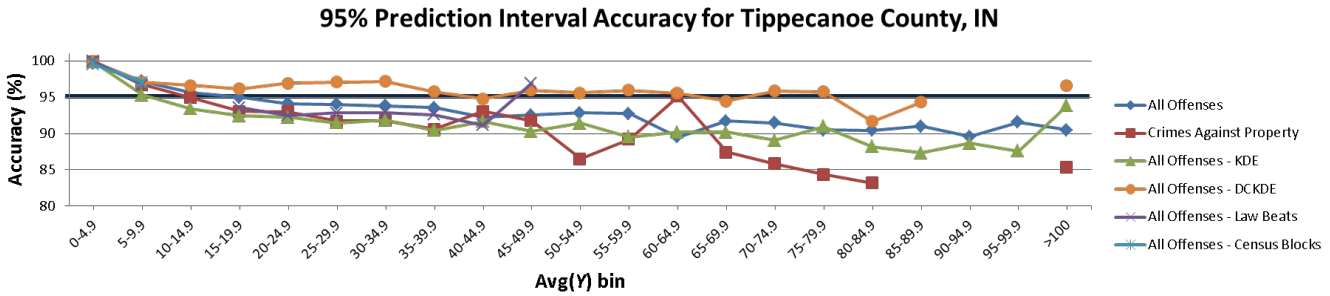


Fig. 6. 95% prediction interval accuracy vs. Avg(Y) for different CTC offenses for Tippecanoe County, IN. Here, geospace has been fragmented into rectangular grids of dimension $k \times k$ ($\forall\, k \in [1,128]$), and by law beats and census blocks.

seasonal (S_v), day-of-the-week (D_v), and remainder variation (R_v) components. However, since we used a weekly aggregation of data, the day-of-the-week component (D_v) must be excluded. Therefore, the time series signal gets modified to $\sqrt{Y_v} = T_v + S_v + R_v$. The prediction step, which involves predicting the value for week $n+1$, remains the same as given in Section 3.

7.2 95% prediction interval accuracy vs. input data average (Avg(Y))

In this method, the geospace was first fragmented into either: (a) rectangular grid regions of dimension $k \times k$ ($\forall\, k \in [1,128]$, with 128 chosen as upper threshold to provide a fine enough geospatial resolution), or (b) man-made geospatial regions (e.g., census blocks, census tracts, law beats, user-specified regions). For each geospatial region, we first generated the incidence count vs. week signal (denote this signal as Y) for a time window of n weeks beginning from the week of, e.g., 1/1/2009. We then used the modified STL forecasting method (Section 7.1) to calculate the 95% prediction interval CI for the predicted week $n+1$, and tested whether the observed data for week $n+1$ fell within the calculated 95% prediction interval for that geospatial region. The average of the input signal Y, $Avg(Y)$, was also calculated.

Next, the input time window was shifted by one week to generate the corresponding incidence count vs. week signal (so, this signal would begin from the week of 1/7/2009). We again computed $Avg(Y)$, and CI for the predicted week $n+1$. As before, we tested whether the observed data for the predicted week $n+1$ fell within the calculated 95% prediction interval. We repeated the process by sliding the time window till it reached the end of available data. For each $Avg(Y)$ value, we maintained two counters that kept track of the number of instances the observed data was within the 95% prediction interval ($C_{Correct}$), and the total instances encountered thus far (C_{Total}). Finally, $Avg(Y)$ values were binned, and $C_{Correct}$ and C_{Total} were summed for each bin. The 95% prediction interval accuracy for each $Avg(Y)$ bin is then given as $\frac{\sum_{bin} C_{Correct}}{\sum_{bin} C_{Total}} \times 100\%$.

7.3 Results and discussion

Figure 6 shows the 95% prediction interval accuracy results for different CTC offenses for Tippecanoe County, IN using the method described in Section 7.2. As can be observed from these results, when the average bin values are low (e.g., less than 10 input samples), the accuracy levels are higher than the expected 95% confidence bound. This indicates that the data are underdispersed for lower input values. In other words, the variance of the observed data is lower than that of the 95% prediction bound when the underlying data are sparse. This conforms to the expected behavior for predicting using our STL forecasting technique: the model predictions get biased if the underlying data are too sparse.

As the input signal average ($Avg(Y)$) values get larger (i.e., more than 10 samples per time step), the prediction accuracy starts to converge at around the expected 95% accuracy level. For example, the prediction interval accuracy for all offenses converges at around 93%. Also, note that the prediction accuracy using the DCKDE method (Section 4.2.2) converges close to the 95% accuracy level; thereby, indicating the efficacy of the technique. It should be noted that since the underlying processes being modeled here (e.g., CTC incidents) are inherently stochastic in nature, perfect 95% confidence bounds will not be achieved (as can be seen from the results in Figure 6). Furthermore, with an uncertain probability distribution of the underlying data, our application of the square root power transform may not guarantee homoscedasticity (i.e., stabilization of variability). This also contributes to our system not achieving perfect 95% confidence bounds. However, even though perfect confidence bounds are not achieved (as can be observed from Figure 6), the accuracy converges close to the 95% bounds. These results show that the underlying data are Normally distributed for higher values of $Avg(Y)$; thereby, satisfying the underlying assumptions of our method used to estimate the 95% confidence interval. This establishes the validity of the claims of our STL prediction methodology in the geospatial domain that the prediction modeling method works as expected as long as the underlying assumptions of the method are satisfied by the data.

Figure 6 shows the 95% prediction interval accuracy vs. input data average results (Section 7.2) for man-made geospatial regions (census blocks and law beats). These results show that the confidence bounds using census blocks are invariably higher than the expected 95% bound, which indicates that the underlying data are underdispersed. Census blocks are small geospatial units, typically bounded by streets or roads (e.g., city block in a city). The smaller $Avg(Y)$ values for census blocks in Tippecanoe County in Figure 6 (less than 10 input samples) further highlight the sparsity of input data. The combination of higher prediction interval accuracy levels and lower $Avg(Y)$ values are telltale for the data sparseness issues we have described, and suggest that the signals generated using census blocks have low predictive statistical power. This further underlines the need to intelligently combine geospatial regions of lower statistical values to obtain a signal of higher predictive power (e.g., as was done in Section 4.3). The 95% prediction interval accuracy results obtained using law beats in Figure 6, on the other hand, shows the accuracy converging at around the expected 95% confidence interval for higher $Avg(Y)$ values (more than 10 input samples). These results provide further evidence that as the underlying data values become larger and begin to conform to the Normal regime, our geospatial prediction methodology provides prediction estimates that are within the expected 95% prediction confidence interval. This further bolsters the applicability and validity of our STL prediction methodology in the geospatial domain.

We also applied the method described in Section 7.2 to all CTC incident category data and generated 95% prediction interval accuracy vs. the input signal average value ($Avg(Y)$) plots for different grid resolutions k. These results are shown in Figure 7. The results indicate that 95% prediction interval accuracy converges at or around the 95% confidence level for large enough $Avg(Y)$ values (i.e., for $Avg(Y)$ bigger than 10). The results indicate that our methodology behaves within the constraints of the Normal regime at higher $Avg(Y)$ values for the different grid dimensions. Also, note that smaller grid dimensions (k) correspond to larger geospatial sub-divisions; and accordingly, smaller k values generate signals of larger counts per bin (i.e., larger $Avg(Y)$

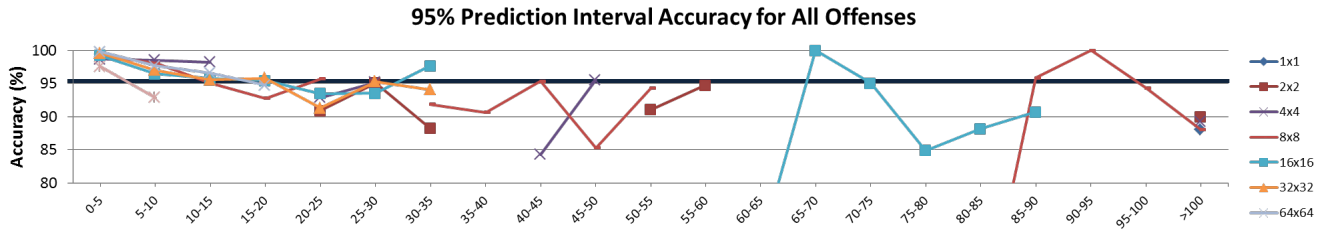


Fig. 7. 95% prediction interval accuracy vs. Avg(Y) for all CTC offenses for Tippecanoe County, IN. Here, geospace has been fragmented into rectangular grids of dimension $k \times k$ for various k values.

values), especially for regions with higher incidence rates. As can be seen from the results in Figures 6 and 7, the accuracy for higher $Avg(Y)$ values tend to be lower than the 95% prediction accuracy; thereby, indicating that the underlying data are slightly overdispersed. These results indicate that coarse scales can generate signals with too much variance, or combinations of multiple signals that overgeneralize the data. Furthermore, the signals generated at coarse scales can be affected by anomalies in underlying data (e.g., crime spikes during unusually high weathers, holidays). These can contribute to the non-Normality of the residuals, and produce an overdispersion of underlying data as compared to the assumptions of our model. It should be noted that although a slight data overdispersion is noticeable at coarse scales, they are deemed to be small enough to currently not warrant any correction. Finally, we note that further research is needed in order to determine the effects of these data overgeneralization issues at coarse scales and to devise strategies to mitigate for their effects.

7.4 Summary

Our model evaluation and validation strategy involved testing for the empirical rule of a Normal distribution where we tested whether the observed data conformed with the 95% prediction interval from our STL forecasting method at various geospatial scales. In order to cope with data sparseness issues, we performed our analysis at a weekly aggregation of data. Our results demonstrate the validity of our approach as long as the underlying assumptions of the underlying models are satisfied by the data. The results obtained using our DCKDE method are also promising. Our results also highlight the importance of performing analysis at appropriate scales, and demonstrate that the model predictions get severely biased when the underlying assumptions are violated by the data. We also explored the effects of data sparseness issues on our model predictions at fine geospatial scales. Our evaluation results show that the model predictions generated using input signals of 10 or more counts per time step on average tend to conform with the 95% prediction confidence intervals. We also highlight the effects of analysis performed at coarse scales, and show the data overgeneralization issues that occur at such scales. Although the results indicate a slight data overdispersion at coarse scales, the results show that the prediction accuracies from the model estimates still tend to converge at around the 95% confidence bounds. This further shows the effectiveness of our forecasting methodology in the geospatial domain. We also note that although our work enables hot spot policing and resource allocation strategy development, further evaluation is required to ascertain the efficacy of our predictive analytics framework when deployed in field. We leave this as future work.

8 Domain Expert Feedback

Our system was assessed by a police captain who oversees the operations and resource allocation of several precincts in a mid-sized police agency (of about 130 sworn officers) in the United States. In this section, we summarize the initial feedback received after conducting several informal interviews with him. The captain emphasized the need for a system that applies a data-driven approach to assist law enforcement decision makers in developing resource allocation strategies. He was impressed by the ability of the system to interactively generate various geospatial and temporal visualizations of historical datasets and forecast maps in real-time. Additionally, he also appreciated having the ability to dynamically apply any desired geospatial, temporal, and/or categorical filters on the data.

The captain stressed the need to carefully combine and aggregate different data categories for which reliable forecast maps could be generated. For example, he noted that a signal generated by combining two crime categories with different attributes (e.g., crimes against property and person) might introduce variability in the forecasting process and produce unreliable results. He further suggested that crimes of opportunity must be filtered out as these exhibit no discernable patterns. He asserted that different regions within the same city can exhibit different crime patterns due to the different underlying region dynamics. He expressed the importance for domain experts to create data category and spatiotemporal templates so viable prediction estimates can be computed using our methodology. Finally, the captain remarked that the predicted hotspot locations using aggregated CTC data occur at the known problem areas in the city.

9 Conclusions and Future Work

In this work, we have presented our visual analytics framework that provides a proactive decision making environment to decision makers and assists them in making informed future decisions using historical datasets. Our approach provides users with a suite of natural scale templates that support analysis at multiple spatiotemporal granularity levels. Our methods are built at the confluence of automated algorithms and interactive visual design spaces that support user guided analytical processes. We enable users to conduct their analyses over appropriate spatiotemporal granularity levels where the scale and frame of reference of the data analysis process and forecasting matches with that of the user's decision making frame of reference. It should be noted that while adjusting for the size of the geospatial and temporal scales is necessary, it is also important to adjust for the scale of the size of the dataset. A forecasting or analysis method that works well for one region with certain demographics and population densities may not have the same efficacy when applied to a different region. As such, our work explores the potential of visual analytics in providing a bridge so that different statistical and machine learning processes occur on the same scale and frame of reference as that of the decision making process.

Our future work includes developing new kernel density estimation techniques designed specifically for improving prediction forecasts. We further plan on improving our designed dynamic covariance kernel density estimation technique (DCKDE) to factor in for temporal distances to further enhance our STL based prediction algorithm. We also plan to incorporate data-driven methods that guide users in selecting between different choices provided by the system based on the underlying features of the data. We also plan on factoring in the influences and correlations among different variables to further refine our natural scale template generation methodology. Finally, we plan on conducting a formal user evaluation in order to understand the efficacy of our system in aiding domain experts to understand the properties of underlying data and their effects on the workings of the different underlying statistical processes.

Acknowledgments

This work was funded by the U.S. Department of Homeland Security VACCINE Center's under Award Number 2009-ST-061-CI0003.

REFERENCES

[1] G. Box and G. Jenkins. *Time series analysis: Forecasting and control*. Holden-Day, San Francisco, 1970.

[2] A. A. Braga. The effects of hot spots policing on crime. *Annals of the American Academy of Political and Social Science*, 578:pp. 104–125, 2001.

[3] A. A. Braga and B. J. Bond. Policing crime and disorder hot spots: A randomized controlled trial*. *Criminology*, 46(3):577–607, 2008.

[4] A. A. Braga, D. M. Hureau, and A. V. Papachristos. An ex post facto evaluation framework for place-based police interventions. *Police Quarterly*, 2012.

[5] C. A. Brewer. *Designing Better Maps: A Guide for GIS users*. ESRI Press, 2005.

[6] D. Brown and R. Oxford. Data mining time series with applications to crime analysis. In *IEEE International Conference on Systems, Man, and Cybernetics*, volume 3, pages 1453–1458 vol.3, 2001.

[7] J. S. d. Bruin, T. K. Cocx, W. A. Kosters, J. F. J. Laros, and J. N. Kok. Data mining approaches to criminal career analysis. In *Proceedings of the Sixth International Conference on Data Mining*, ICDM '06, pages 171–177, Washington, DC, USA, 2006. IEEE Computer Society.

[8] D. V. Canter. The environmental range of serial rapists. In D. V. Canter, editor, *Psychology in Action*, Dartmouth Benchmark Series, pages 217–230. Dartmouth Publishing Company, Hantshire, UK, January 1996.

[9] R. B. Cleveland, W. S. Cleveland, J. E. McRae, and I. Terpenning. Stl: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, 6:3–73, 1990.

[10] G. Cowan. *Statistical data analysis*. Oxford University Press, 1998.

[11] P. J. Diggle and P. J. Diggle. Statistical analysis of spatial point patterns. *London: Edward Arnold*, 1983.

[12] P. J. Diggle, J. Tawn, and R. Moyeed. Model-based geostatistics. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 47(3):299–350, 1998.

[13] G. Farrell and K. Pease. *Repeat victimization*, volume 12. Criminal Justice Press, 2001.

[14] M. Felson and E. Poulsen. Simple indicators of crime by time of day. *International Journal of Forecasting*, 19(4):595–601, 00 2003.

[15] J. S. Goldkamp and E. R. Vilcica. Targeted enforcement and adverse system side effects: The generation of fugitives in philadelphia. *Criminology*, 46(2):371–409, 2008.

[16] R. Hafen, D. Anderson, W. Cleveland, R. Maciejewski, D. Ebert, A. Abusalah, M. Yakout, M. Ouzzani, and S. Grannis. Syndromic surveillance: Stl for modeling, visualizing, and monitoring disease counts. *BMC Medical Informatics and Decision Making*, 9(1):21, 2009.

[17] K. Harries. *Crime and the environment*. American Lecture Series; No. 1033. Charles C. Thomas Publisher, Limited, 1980.

[18] S. D. Johnson, W. Bernasco, K. J. Bowers, H. Elffers, J. Ratcliffe, G. Rengert, and M. Townsley. Space–time patterns of risk: a cross national assessment of residential burglary victimization. *Journal of Quantitative Criminology*, 23(3):201–219, 2007.

[19] M. H. Kutner, C. J. Nachtsheim, J. Neter, and W. Li. *Applied linear statistical models*, volume 5. McGraw-Hill Irwin Chicago, 2004.

[20] R. Maciejewski, R. Hafen, S. Rudolph, S. Larew, M. Mitchell, W. Cleveland, and D. Ebert. Forecasting hotspots: A predictive analytics approach. *IEEE Transactions on Visualization and Computer Graphics*, 17(4):440 –453, April 2011.

[21] A. Malik, R. Maciejewski, T. F. Collins, and D. S. Ebert. Visual analytics law enforcement toolkit. In *IEEE International Conference on Technologies for Homeland Security*, pages 222–228, 2010.

[22] A. Malik, R. Maciejewski, N. Elmqvist, Y. Jang, D. Ebert, and W. Huang. A correlative analysis process in a visual analytics environment. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 33–42, 2012.

[23] M. Monroe, R. Lan, H. Lee, C. Plaisant, and B. Shneiderman. Temporal event sequence simplification. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2227–2236, 2013.

[24] J. D. Morenoff, R. J. Sampson, and S. W. Raudenbush. Neighborhood inequality, collective efficacy, and the spatial dynamics of urban violence*. *Criminology*, 39(3):517–558, 2001.

[25] T. Muhlbacher and H. Piringer. A partition-based framework for building and validating regression models. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):1962–1971, Dec 2013.

[26] D. A. Norman. *Things that make us smart: Defending human attributes in the age of the machine*. Basic Books, 1993.

[27] R. Oppenheim. Forecasting via the box-jenkins method. *Journal of the Academy of Marketing Science*, 6(3):206–221, 1978.

[28] A. Rind, T. Lammarsch, W. Aigner, B. Alsallakh, and S. Miksch. Timebench: A data model and software library for visual analytics of time-oriented data. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2247–2256, 2013.

[29] G. Robertson, D. Ebert, S. Eick, D. Keim, and K. Joy. Scale and complexity in visual analytics. *Information Visualization*, 8(4):247–253, 2009.

[30] L. W. Sherman. Police crackdowns: Initial and residual deterrence. *Crime and Justice*, 12:pp. 1–48, 1990.

[31] L. W. Sherman, P. R. Gartin, and M. E. Buerger. Hot spots of predatory crime: Routine activities and the criminology of place. *Criminology*, 27(1):27–56, 1989.

[32] M. Short, M. Dorsogna, P. Brantingham, and G. Tita. Measuring and modeling repeat and near-repeat burglary effects. *Journal of Quantitative Criminology*, 25(3):325–339, 2009.

[33] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall/CRC, 1986.

[34] S. J. South and S. F. Messner. Crime and demography: Multiple linkages, reciprocal relations. *Annual Review of Sociology*, 26(1):83–106, 2000.

[35] J. Stasko, C. Görg, and Z. Liu. Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization*, 7(2):118–132, 4 2008.

[36] S. S. Stevens. On the theory of scales of measurement, 1946.

[37] J. J. Thomas and K. A. Cook, editors. *Illuminating the Path: The R&D Agenda for Visual Analytics*. IEEE Press, 2005.

[38] W. R. Tobler. A computer movie simulating urban growth in the detroit region. *Economic geography*, pages 234–240, 1970.

[39] S. Towers. Kernel probability density estimation methods. *Proceedings of the Advanced Statistical Techniques in Particle Physics*, pages 107–111, 2002.

[40] P. F. Velleman and L. Wilkinson. Nominal, ordinal, interval, and ratio typologies are misleading. *The American Statistician*, 47(1):65–72, 1993.

[41] D. Weisburd, J. Hinkle, C. Famega, and J. Ready. The possible backfire effects of hot spots policing: an experimental assessment of impacts on legitimacy, fear and collective efficacy. *Journal of Experimental Criminology*, 7(4):297–320, 2011.

[42] L. Wilkinson and G. Wills. *The Grammar of Graphics*. Statistics and Computing. Springer, 2005.

[43] P. C. Wong, L. R. Leung, N. Lu, M. Paget, J. C. Jr., W. Jiang, P. Mackey, Z. T. Taylor, Y. Xie, J. Xu, S. Unwin, and A. Sanfilippo. Predicting the impact of climate change on u.s. power grids and its wider implications on national security. In *AAAI Spring Symposium: Technosocial Predictive Analytics*, pages 148–153. AAAI, 2009.

[44] C.-H. Yu, M. W. Ward, M. Morabito, and W. Ding. Crime forecasting using data mining techniques. In *Proceedings of the IEEE 11th International Conference on Data Mining Workshops*, ICDMW '11, pages 779–786, Washington, DC, USA, 2011. IEEE Computer Society.

[45] J. Yue, A. Raja, D. Liu, X. Wang, and W. Ribarsky. A blackboard-based approach towards predictive analytics. In *AAAI Spring Symposium: Technosocial Predictive Analytics*, page 154. AAAI, 2009.

A Framework for Hierarchical Ensemble Clustering

LI ZHENG, School of Computer Science and Engineering, Nanjing University of Science and Technology; and School of Computer Science, Florida International University
TAO LI, School of Computer Science, Florida International University
CHRIS DING, Department of Computer Science, University of Texas at Arlington

Ensemble clustering, as an important extension of the clustering problem, refers to the problem of combining different (input) clusterings of a given dataset to generate a final (consensus) clustering that is a better fit in some sense than existing clusterings. Over the past few years, many ensemble clustering approaches have been developed. However, most of them are designed for partitional clustering methods, and few research efforts have been reported for ensemble hierarchical clustering methods. In this article, a hierarchical ensemble clustering framework that can naturally combine both partitional clustering and hierarchical clustering results is proposed. In addition, a novel method for learning the ultra-metric distance from the aggregated distance matrices and generating final hierarchical clustering with enhanced cluster separation is developed based on the ultra-metric distance for hierarchical clustering. We study three important problems: dendrogram description, dendrogram combination, and dendrogram selection. We develop two approaches for dendrogram selection based on tree distances, and we investigate various dendrogram distances for representing dendrograms. We provide a systematic empirical study of the ensemble hierarchical clustering problem. Experimental results demonstrate the effectiveness of our proposed approaches.

Categories and Subject Descriptors: H.3.3 [**Information Search and Retrieval**]: Clustering; I.2.6 [**Artificial Intelligence**]: Learning

General Terms: Design, Algorithms, Performance

Additional Key Words and Phrases: Hierarchical ensemble clustering, ultra-metric, ensemble selection

ACM Reference Format:
Li Zheng, Tao Li, and Chris Ding. 2014. A framework for hierarchical ensemble clustering. ACM Trans. Knowl. Discov. Data 9, 2, Article 9 (September 2014), 23 pages.
DOI: http://dx.doi.org/10.1145/2611380

1. INTRODUCTION

Data clustering arises in many disciplines and has a wide range of applications. The general goal of data clustering is to group a finite set of points in a multidimensional space into clusters so that points in the same cluster are similar to each other, whereas points in different clusters are dissimilar. The clustering problem has been extensively

This work is partially supported by the U.S. Department of Homeland Security under grant Award Number 2010-ST-062-000039; the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CIO001; the National Science Foundation under grants DBI-0850203, HRD-0833093, and DMS-0915110; and the Army Research Office under grants number W911NF-10-1-0366 and W911NF-12-1-0431.
Author's address: L. Zheng and T. Li, School of Computer Science, Nanjing University of Science and Technology, Nanjing 210094, China; School of Computer Science, Florida International University, 11200 SW 8th ST, Miami, FL 33199; email: {lzhen001, taoli}@cs.fiu.edu; C. Ding, Department of Computer Science and Engineering, University of Texas at Arlington, Arlington, TX, 76019; email: chqding@uta.edu.

DOI: http://dx.doi.org/10.1145/2611380

studied in the data mining, database, and machine learning communities, and many different approaches have been developed from various perspectives with various focuses. Based on the way the clusters are generated, these clustering methods can be roughly divided into two categories: partitional clustering and hierarchical clustering [Tan et al. 2005]. Generally, **partitional clustering** decomposes the dataset into a number of disjoint clusters that typically represent a local optimum of some predefined objective functions. Hierarchical clustering groups the data points into a hierarchical tree structure using bottom-up or top-down approaches. Also, equivalent dendrogram representation can be generated based on metric fitting.

Clustering is an inherently difficult problem. Different clustering algorithms and even multiple trials of the same algorithm may produce different results due to random initializations and stochastic learning methods. Recently, ensemble clustering has emerged as an important extension of the classical clustering problem because it can overcome the resulting instability and improve clustering performance. It refers to the following problem: Given a number of different (input) clusterings that have been generated for a dataset, find a single final (consensus) clustering that is a better fit in some sense than the existing clusterings [Strehl and Ghosh 2003]. Over the past few years, many ensemble clustering techniques have been proposed [Li et al. 2007, 2004 Azimi and Fern 2009; Fern and Brodley 2004; Gionis et al. 2005; Li and Ding 2008; Monti et al. 2003; Topchy et al. 2005; Luo et al. 2011].

However, existing ensemble techniques are primarily designed for partitional methods, and few research efforts have been reported for ensemble hierarchical clustering methods. In partitional clustering, the clustering results are "flat" and can be easily represented using vectors, clustering indicators, or connectivity matrices [Li and Ding 2008; Strehl and Ghosh 2003]. Different from partitional clustering, hierarchical clustering results are often more complex, and they are typically represented as dendrograms or trees.

In this work, we propose a novel **Hierarchical Ensemble Clustering** (HEC) framework in which the input can be both partitional clusterings and hierarchical clusterings. The output of the framework is a **consensus** hierarchical clustering. Three different cases are described here.

(1) In this case, the input clusterings are partitional clusterings. The **aggregate consensus distance** from these partitional clusterings is first constructed, and a consensus clustering using the consensus distance is then generated. These steps lead to the usual ensemble clustering. In HEC, a **structure hierarchy** can be further generated on top of the consensus clustering using the consensus distance.

Note that a structure hierarchy on top of a clustering solution is useful to organize and understand the discovered knowledge (topic or pattern). In addition, the cluster structure hierarchy resolves a problem in the usual ensemble clustering when the input partitional clusterings have different number of clusters.

In this case, K, the number of clusters in the final clustering solution, is not uniquely determined (much research has been done on finding the most appropriate number of clusters in a dataset [Fraley and Raftery 1998; Sugar and James 2003; Tibshirani et al. 2001]). In ensemble clustering, we consider input partitional clusterings, including the **number of clusters** in each input partitional clustering, as meaningful results. Therefore, if the number of clusters of input partitional clusterings has a range of $[K_1, K_2]$, then the number of clusters in the final ensemble clustering should be $K \in [K_1, K_2]$. From this analysis, in the HEC framework, we can set $K = K_2$ for the bottom clusterings (leaves) of the structure hierarchy. In this way, the "true" number of clusters is guaranteed to be inside the cluster structure hierarchy.

(2) In this case, the input clusterings are hierarchical clusterings (i.e., a set of dendrograms). A dendrogram is defined to be nested family of partitions, usually represented

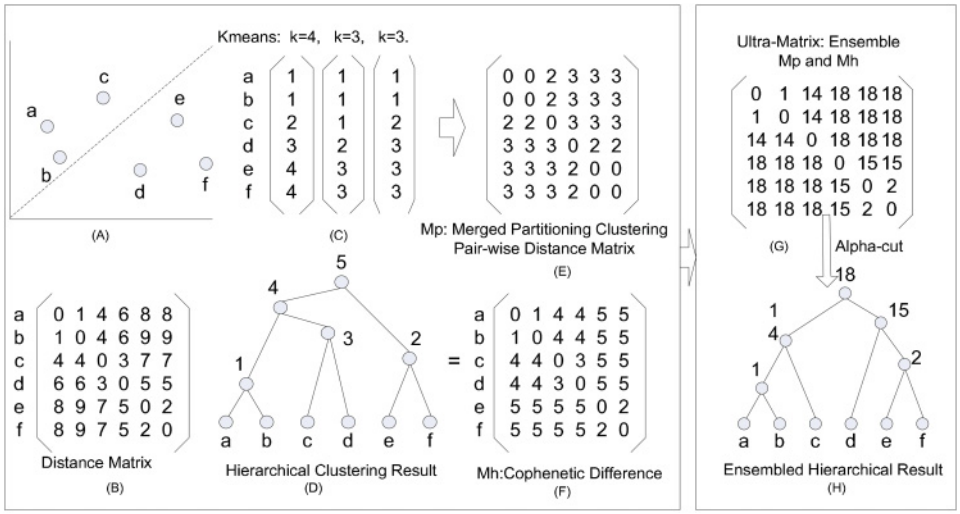


Fig. 1. An illustrative example of hierarchical ensemble clustering with both partitional and hierarchical clusterings as input. The dataset is shown in (A), and their distances are shown in (B). K-means clustering are performed in (C) and lead to a consensus distance matrix in (E). A hierarchical clustering is done in (D) and leads to a dendrogram distance matrix in (F). The consensus distance matrix of (E) and the dendrogram distance matrix in (F) are combined in (G), and the final hierarchical clustering are generated in (H).

graphically as a rooted tree [Podani 2000]. Dendrograms are often used to represent a hierarchical decomposition of the underlying data set.

The **aggregate dendrogram distance** is first constructed between objects and then a hierarchical clustering as the final solution is generated as the final solution.

(3) In this case, the input clusterings contain both partitional clusterings and hierarchical clusterings. The consensus distance from the partitional clusterings and the dendrogram distance from hierarchical clusterings are first constructed. These two distances into are then combined into a single distance, and a hierarchical clustering is generated as the final solution. An illustrative example is shown in Figure 1. Figure 1(A) shows the example dataset and Figure 1(B) shows the distance matrix. K-means clustering results with different numbers of clusters are presented in Figure 1(C) and lead to a consensus distance matrix shown in Figure 1(E). A hierarchical clustering is performed in Figure 1(D) and generates a dendrogram distance matrix shown in Figure 1(F). The consensus distance matrix of Figure 1(E) and the dendrogram distance matrix in Figure 1(F) are combined in Figure 1(G), and the final hierarchical clustering is generated in Figure 1(H).

Our preliminary work was presented at the International Conference on Data Mining (ICDM) 2010 [Zheng et al. 2010] in which we focused on the ensembles of hierarchical clustering and the related computational algorithms. In this journal article, we extend our previous work by systematically studying the following three important problems:

(1) ***Dendrogram Description***: How can we represent the dendrograms so that different hierarchical clustering solutions can be compared and combined?
(2) ***Dendrogram Combination***: How can we aggregate different dendrograms and generate final hierarchical solution?
(3) ***Dendrogram Selection***: Given a large collection of input hierarchical clusterings, how can we select a subset from the input collection to effectively build an ensemble solution that performs as well as or even better than using all available clusterings [Fern and Lin 2008]?

In particular, we investigate various descriptor matrices for representing dendrograms and propose a novel method for deriving a final hierarchical clustering by fitting an ultra-metric from the aggregated descriptor matrix. Here, we study the problem of combining both hierarchical and partitional clustering results, whereas our conference paper only focuses on the combination hierarchical clusterings. In this journal article, we present a method to first represent multiple partitional clustering results as a distance matrix and then effectively combine it with dendrogram descriptors. Thus, the final dendrogram naturally takes both types of clustering results into consideration. We formalize the ultra-metric transformation problem as an optimization problem and prove the correctness of our solution. This article also studies the problem of ensemble selection, which was ignored in our conference paper. The dendrogram selection mechanism, considering both the quality and the diversity of individual hierarchical clustering results, is presented and two approaches for dendrogram selection based on tree distances are developed. In addition, more experimental results, including using large datasets and different hierarchical clustering methods with different sets of base clusterings, are reported this article. Our experimental evaluation also provides a systematic empirical study on the ensemble hierarchical clustering problem. Experimental results have demonstrated the effectiveness of our proposed approaches.

The rest of the article is organized as follows: Section 2 discusses the related work; Section 3 discusses the ultra-metric and the general algorithm strategy for hierarchical ensemble clustering; Section 4 investigates various descriptor matrices for representing dendrograms; Section 5 describes the distance matrix used for representing partitional clustering results; Section 6 proposes a novel method for deriving final hierarchical clustering by fitting an ultra-metric from the aggregated distance matrix; Section 7 presents our approaches for dendrogram selection (i.e., selecting a subset of hierarchical clusterings from the input collection); Section 8 shows experimental evaluations and result analysis; and, finally, Section 9 concludes the paper and discusses future work.

2. RELATED WORK

2.1. Hierarchical Clustering

Hierarchical clustering algorithms are unsupervised methods to generate tree-like clustering solutions. They group the data points into a hierarchical tree structure using bottom-up (agglomerative) or top-down (divisive) approaches [Tan et al. 2005]. The typical bottom-up approach takes each data point as a single cluster to start with and then builds bigger clusters by grouping similar data points together until the entire dataset is encapsulated into one final cluster. The divisive approaches start with all data points in one cluster and then split the larger clusters recursively. Many research efforts have been reported on algorithm-level improvements to the hierarchical clustering process and on understanding hierarchical clustering [Wu et al. 2009; Zhao and Karypis 2002; Zheng and Li 2011].

2.2. Ensemble Clustering

Ensemble clustering refers to the problem of finding a combined clustering result based on multiple input clusterings of a given dataset. Many techniques can be used to obtain multiple clusterings, such as applying different clustering algorithms, using resampling to get subsamples of the dataset, utilizing feature selection methods to obtain different feature spaces, and exploiting the randomness of the clustering algorithm. Many approaches have been developed to solve ensemble clustering problems over the past few years [Azimi and Fern 2009; Fern and Brodley 2004; Gionis et al. 2005; Li and Ding 2008; Monti et al. 2003; Topchy et al. 2005]. However, existing ensemble clustering techniques are mainly designed for partitional clustering methods. The

problem of ensemble hierarchical clustering using dendrogram descriptors has been studied in Mirzaei et al. [2008]. The key difference here is that we present a coherent algorithm to learn the closest ultra-metric solution (matrix B in Equation (6)) whereas the approach in Mirzaei et al. [2008] requires many parameters that are selected in an ad hoc manner. In our approach, there are no parameters. In addition, we propose a hierarchical ensemble clustering framework that can naturally combine both partitional clustering and hierarchical clustering results, and we systematically study the problems related to dendrogram description, selection, and combination.

2.3. Consensus Tree

The problem of finding the consensus tree has been extensively studied in bioinformatics when comparing the evolution of species to reach a consensus or agreement [Adams 1986; Adams 1972]. Most techniques for solving the problem are based on agreement subtrees (e.g., the substructures that are common to all the trees) [Farach et al. 1995; Wilkinson 1994]. It is quite difficult for these consensus tree techniques to preserve structural information while including all the existing leaves from the input trees [Swofford 1991]. In our work, a framework based on descriptor matrices is proposed to preserve the common structures from the input clusterings and generate a full consensus tree.

2.4. Metric Fitting

Fitting a tree metric to the (dis-)similarity data has been studied quite extensively [Ailon and Charikar 2005]. Ultra-metric is a special kind of tree metric in which all elements of the input dataset are leaves in the underlying tree, and all leaves are at the same distance from the root. It naturally corresponds to a hierarchy of clusterings [Agarwala et al. 1999; Ailon and Charikar 2005]. Given a dissimilarity D on pairs of objects, the problem of finding the best ultra-metric d_u such that $||D-d_u||_p$ is minimized is NP-hard for L_1 and L_2 norms (e.g., when $p = 1$ and $p = 2$) [Agarwala et al. 1999]. In our work, a new method for fitting an ultra-metric to the aggregated descriptor matrix is developed.

2.5. Ensemble Decision Trees

In supervised classification, different decision trees can be combined using bagging [Breiman and Breiman 1996], boosting [Schapire and Singer 1999], stacking [Wolpert 1992], or random forests [Breiman and Breiman 2001]. Unlike our ensemble hierarchical clustering, these ensemble methods are designed for supervised classification. In addition, most of the decision tree ensembles do not generate a final tree and simply combine the output predictions of base trees.

2.6. Cluster Ensemble Selection

The problem of selecting a subset of input clusterings to form a smaller but better performing cluster ensemble than using all available solutions has been studied recently for partitional clustering [Azimi and Fern 2009; Fern and Lin 2008]. In this article, we develop cluster ensemble selection methods for hierarchical clustering based on tree distances.

There are also many related researches on combining multiple hierarchical clustering results from different perspectives [Hossain et al. 2012; Jalalat-evakilkandi and Mirzaei 2010; Koutroumbas et al. 2010; Lu and Wan 2012; Mirzaei and Rahmati 2008; Mirzaei and Rahmati 2010; Rashedi and Mirzaei 2011]. However, our proposed approach in this article is able to combine both multiple hierarchical clustering and partitional clustering results. In addition, we studied the problem of dendrogram

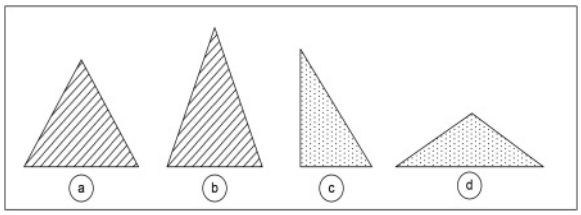


Fig. 2. A ultra-metric space example.

selection and also developed a method for learning the ultra-metric distance from the aggregated distance.

3. ULTRA-METRIC AND DENDROGRAM RECONSTRUCTION

A dendrogram is defined as a nested family of partitions, usually represented graphically as a rooted tree where leaves represent data objects and internal nodes represent clusters at various levels [Podani 2000]. The structural information is kept by pairwise cophenetic proximity that measures the level at which two data objects are first merged into a cluster [Jain and Dubes 1998].

Given a dendrogram, our task is to assign distances between leaf nodes. This problem has been studied in the literature [Mirzaei et al. 2008; Podani 2000]. Several commonly used dendrogram distances (also called descriptors) are described in Section 4. Note that each of these dendrogram distance is in fact an ultra-metric distance. This is important because given an ultra-metric distance matrix $D = (d_{ij})$, we can reconstruct the original tree.

3.1. Ultra-metric Distance

Definition 1. A distance matrix $D = (d_{ij})$ is a **metric**, if it has the following properties: (1) nonnegativity

$$d_{ij} \geq 0,$$

if $d_{ij} = d(x_i, x_j) = 0$, then $x_i = x_j$; (2) symmetry

$$d_{ij} = d_{ji};$$

and (3) the **triangle inequality**

$$d_{ij} \geq 0, \quad d_{ij} \leq d_{ik} + d_{kj}, \ i \neq k \neq j.$$

Although non-negativity and symmetry hold for many distance measures in data mining, the triangle inequality often does not always hold. A more restricted version of the triangle inequality is called the **ultra-metric inequality**:

$$d_{ij} \leq max(d_{ik}, d_{jk}) \tag{1}$$

for all triplets of points i, j, k. This is equivalent to saying that for any distinct triple i, j, k, the largest two distances among d_{ij}, d_{ik}, d_{jk} are equal and not less than the third one.

Definition 2. A distance measure is an ultra-metric if it satisfies the ultra-metric inequality, non-negativity, and symmetry.

To illustrate the ultra-metric, four triangles formed by three data points are shown in Figure 2. Those four triangles clearly satisfy the triangle inequality; however, only a and b satisfy the ultra-metric inequality. From Equation (1), it can be easily shown that, for those triangles shown in Figure 2, if the proximity measure is an ultra-metric,

then the triangle formed by all triples of points must be an isosceles triangle with the unequal leg no longer than the two legs of equal length. The example shows that ultra-metric properties impose more restrictions on sample relations.

A distance measure automatically satisfies the triangle inequality if it satisfies the ultra-metric inequality. Thus, an ultra-metric distance is also a metric distance; but the converse is not true.

3.2. Dendrogram Reconstruction and Ultra-metric

In Single-Link (SL) and Complete-Link (CL) hierarchical clustering, a dendrogram is generated by repeatedly picking the closest pair of clusters from the distance matrix, merging these two clusters into one, and updating the distance matrix. Various schemes differ in how the distance between a newly formed cluster and the other clusters is defined. Let d be the final generated distance. It can be easily shown that d is an ultra-metric. To see why, consider three objects i, j, k. Without loss of generality, assume i and j merge first. Then we have $d(i, j) \leq d(i, k) = d(j, k)$. More details can be found in Jain and Dubes [1998].

In our HEC framework, ultra-metric distance plays a critical role due to its unique reconstruction property. We have the following proposition:

PROPOSITION 1. *From a given ultra-metric distance D, a unique dendrogram G can be constructed, in the sense that if we construct the distance from G, we recover D exactly.*

In fact, there are several ways to model the pairwise distance matrix between instances in a dendrogram (see Section 4). Using different dendrogram distance measures leads to different ultra-metric distances.

3.3. Hierarchical Ensemble Clustering Algorithm Strategy

With the aforementioned discussions on ultra-metric distances and dendrograms, the algorithmic strategy of our hierarchical ensemble clustering is outlined here:

(1) Use a dendrogram distance measure to generate an ultra-metric dendrogram distance for each input dendrogram (see Section 4). We also discuss the consensus distance matrix for partitional clustering results in Section 5.
(2) Aggregate the ultra-metric dendrogram distances, as well as the consensus distance for partitional clusterings (see Section 6).
(3) Find the closest ultra-metric distance from the aggregated distance (see Section 6).
(4) Construct the final hierarchical clustering (see Section 6).

4. DENDROGRAM DISTANCES

A dendrogram is usually used to represent the hierarchical clustering results for cluster analysis, and it is easy to interpret. The ultra-metric information contained in the pairwise distance matrix can be clearly mapped to dendrogram structural information. So, for each dendrogram, there is an ultra-metric matrix that uniquely characterizes it and can be used to recover this dendrogram [Mirzaei et al. 2008].

For instance, a dendrogram obtained from the SL hierarchical clustering algorithm can be viewed as a weighted dendrogram in which every internal node is associated with a continuous variable indicating the merge distance within all its covered leaves. The merge distance is usually called the *height*. If we replace the height of an internal node with its rank order (i.e., the *level*), which is maintained globally with respect to the whole dendrogram, then a weighted dendrogram becomes a fully ranked dendrogram [Podani 2000]. A dendrogram descriptor can be viewed as a distance function describing the relative position of a given pair of leaves in the dendrogram, and it is used to characterize a corresponding dendrogram.

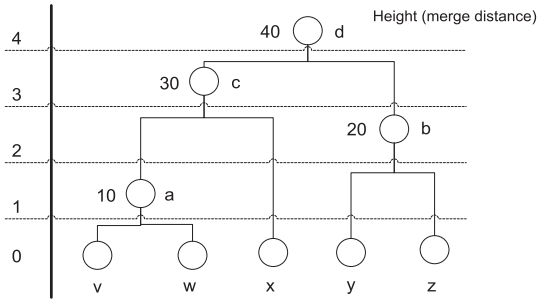


Fig. 3. A dendrogram example.

In the following paragraphs, we introduce several dendrogram descriptors used in our work. The first three dendrogram descriptors are based on a fully ranked dendrogram, and they all make use of the level information [Mirzaei et al. 2008; Podani 2000]. In other descriptors, the level information is not directly considered.

—***Cophenetic Difference*** **(CD)**: the lowest height (i.e., merge distance) of internal nodes in the dendrogram where two specified leaves are joined together. For example, CD between nodes v and x in Figure 3 is 30.
—***Maximum Edge Distance*** **(MED)**: the depth of a node in a bottom-up view. All leaf nodes are assigned a depth of 0, and the depth of any internal node is generated in a bottom-up manner. Suppose C3 is the internal node at which C1 and C2 first merge; then, $\text{Depth}(C3) = \max(\text{Depth}(C1), \text{Depth}(C2)) + 1$. For example, MED of nodes v and x in Figure 3 is 2. Nodes v and x first merged at internal node c, so $\text{Depth}(c) = \max(\text{Depth}(a), \text{Depth}(x)) + 1 = \max(1, 0) + 1 = 2$, since $\text{Depth}(a) = \max(\text{Depth}(v), \text{Depth}(w)) + 1 = 1$.
—***Partition Membership Divergence*** **(PMD)**: PMD utilizes the property that a hierarchical clustering result implies a sequence of nested partitions and is defined as the number of partitions of the hierarchy in which two specified leaves are not in the same cluster.
—***Cluster Membership Divergence*** **(CMD)**: the size of the smallest cluster in the hierarchy that contains two specified leaves.
—***Subdendrogram Membership Divergence*** **(SMD)**: the number of subdendrograms in which two specified leaves are not included together.

For illustration purpose, an example dendrogram is given in Figure 3, and its various descriptor matrices are presented in Table I.

5. DISTANCE MATRICES FOR PARTITIONAL CLUSTERING RESULTS

As discussed in Section 1, our framework can be naturally extended to ensemble both partitional and hierarchical clustering results by representing the partitional clustering results with a distance matrix.

Formally let $X = \{x_1, x_2, \ldots, x_n\}$ be a set of n data points. Given a partitional clustering C consisting of a set of clusters $C = \{C_1, C_2, \ldots, C_k\}$ where k is the number of clusters and $X = \bigcup_{\ell=1}^{k} C_\ell$, we can define the following associated distance matrix $D(C)$ whose ij-th entry is defined as

$$d_{ij} = \begin{cases} 0 & (i, j) \in C_\ell \\ 1 & \text{Otherwise,} \end{cases} \tag{2}$$

Table I. Dendrogram Descriptors for the Sample Dendrogram in Figure 3

1: CD

	v	w	x	y	z
v	0	10	30	40	40
w	10	0	30	40	40
x	30	30	0	40	40
y	40	40	40	0	20
z	40	40	40	20	0

2: CMD

	v	w	x	y	z
v	1	2	3	5	5
w	2	1	3	5	5
x	3	3	1	5	5
y	5	5	5	1	2
z	5	5	5	2	1

3:MED

	v	w	x	y	z
v	0	1	2	3	3
w	1	0	2	3	3
x	2	2	0	3	3
y	3	3	3	0	1
z	3	3	3	1	0

4: PMD

	v	w	x	y	z
v	0	1	3	4	4
w	1	0	3	4	4
x	3	3	0	4	4
y	4	4	4	0	2
z	4	4	4	2	0

5:SMD

	v	w	x	y	z
v	1	1	2	3	3
w	1	1	2	3	3
x	2	2	2	3	3
y	3	3	3	2	2
z	3	3	3	2	2

where $(i, j) \in C_\ell$ means that i-th data point and j-th data point are in the same cluster C_ℓ. In other words, if the i-th data point and the j-th data point are in the same cluster, then the distance between them is 0.

Given a set of s clusterings (or partitions) $\mathcal{P} = \{P^1, P^2, \ldots, P^s\}$ of the data points in X, the associated consensus distance matrix D can be represented as

$$D(\mathcal{P}) = \frac{1}{s} \sum_{i=1}^{s} D(P^i). \tag{3}$$

In other words, the ij-th entry of D indicates the average number of times that the i-th data point and the j-th data point are not in the same cluster.

Equation (3) defines a way to aggregate multiple partitional clustering results into one consensus distance matrix. Also there are many different ways to define the consensus function, such as co-associations between data points or based on pairwise agreements between partitions. Some of the criteria are based on the similarity between data points, and some of them are based on the estimates of similarity between partitions. The relationship between consensus matrix and other measures is discussed and summarized in Li et al. [2010].

Note that the distance matrix can be combined with the dendrogram descriptors to form the aggregated distance matrix for dendrogram combination. A weight can be assigned to the distance matrix to ensure that it is at the same scale as the dendrogram descriptors.

6. DENDROGRAM COMBINATION

Given any similarity, we can do any kind of hierarchical clustering. However, there are many different choices here: SL, CL, average-link, and many other choices. Which one to choose? Our logic is that since the input individual descriptors are ultra-metric, and the consensus matrix is not ultra-metric, the most natural approach is to a find an ultra-metric that is as close to the consensus matrix as possible. Once this ultra-metric

is learned, the final hierarchical clustering is uniquely determined. There are other choices here. The entire approach is uniquely deterministic.

Let $D(\mathcal{P})$ be the computed consensus distance from the input partitional clusterings and let $D(H)$ be the aggregated dendrogram distance from the input hierarchical clusterings. The task of dendrogram combination includes the following steps:

(1) Finding an ultra-metric distance T which is the closest to $D = \frac{1}{2} \times (D(\mathcal{P}) + D(H))$
(2) Constructing the final hierarchical clustering based on T

Once the ultra-metric T is obtained, the final hierarchical clustering can be generated by performing the alpha-cut [Meyer et al. 2004]. In the remainder of this section, we concentrate on (1); that is, how to compute T.

It should be pointed out that the aggregated distance D will not be ultra-metric, even if each individual dendrogram distance is an ultra-metric. We compute the ultra-metric distance T that is closest to D, instead of using D directly, due to the following two reasons. The first reason is for the unique reconstruction of the eventual dendrogram, the final hierarchical clustering, as discussed in Section 3. The second reason is that we can use a transitive dissimilarity to construct T that could attract nearby data objects into a closer proximity.

6.1. Transitive Dissimilarity

Our task is to construct the transitive dissimilarity starting from D. Note that the nonnegative distance D can be viewed as the edge weight on a graph.

The idea of transitive dissimilarity is to **preserve the transitivity** of a graph; more precisely, a social network with n people represented as $(V_1 \ldots, V_n)$. If person V_1 knows person V_2, and person V_2 knows person V_3, transitivity implies that person V_1 knows person V_3. Turning this into distances, the transitivity of $V_1 \to V_2 \to V_3$ can be enforced as

$$d_{13} \leq \max(d_{12}, d_{23}),$$

that is, the distance d_{13} should be no greater than either d_{12} or d_{23}.

Now consider four people. One can see that our enforcement satisfies associativity: If both $d_{13} \leq \max(d_{12}, d_{23})$ and $d_{24} \leq \max(d_{23}, d_{34})$ hold, then

$$d_{14} \leq \max(d_{12}, d_{23}, d_{34}).$$

Generalizing to any path P_{ij} between i and j, on the graph, the **transitive dissimilarity** on a path P_{ij} (a set of edges connect V_i and V_j) can be defined as

$$T(P_{ij}) = \max(d_{i,k_1}, d_{k_1,k_2}, d_{k_2,k_3}, \ldots, d_{k_{n-1},k_n}, d_{k_n,j}). \tag{4}$$

So, for any given pair of vertices V_i and V_j, the transitive dissimilarity varies according to different paths chosen between V_i and V_j. The **minimal transitive dissimilarity** is defined as:

$$m_{ij} = \min_{P_{ij}}(T(P_{ij})), \text{ for given vertices } V_i \text{ and } V_j. \tag{5}$$

It is clear that $m_{ij} \leq d_{ij}, \forall V_i$ and V_j, which implies that minimal transitive dissimilarity brings vertices closer than the original distance matrix.

Thus, the problem of obtaining the ultra-metric transformation of a consensus matrix can be formulated as the following optimization problem:

PROBLEM 1. *A is the consensus distance matrix; B is the desired ultra-metric to be computed:*

$$\min_B \sum_{ij} |A_{ij} - B_{ij}|, s.t. \;\; B_{ij} \leq A_{ij}. \tag{6}$$

The ultra-metric constraint on B is a hard constraint. The optimal solution is given by Algorithm 1. In other words, the desired ultra-metric distance is always smaller than input distance.

ALGORITHM 1: Modified Floyd-Warshall Algorithm to Compute the Minimum Transitive Dissimilarity of Weighted Graph G

Input: G: Pairwise distance matrix of dataset.
Output: M: Minimum transitive dissimilarity matrix closure of G.
Init: M = G.

```
for k ← 0 to N do
  for i ← 0 to N do
    for j ← 0 to N do
      m_ij = min(m_ij, max(m_ik, m_kj))
    end for
  end for
end for
return M
```

The modified Floyd-Warshall algorithm [Ding et al. 2006] is used to compute the updated transitive dissimilarity of all pairs of vertices in the weighted graph. Algorithm 1 describes the algorithm procedure where the adjacency matrix G of a weighted graph with N nodes is given as the input.

The following propositions are needed to show the correctness of the modified Floyd-Warshall algorithm.

PROPOSITION 2. *Suppose the edge weights of a given graph satisfy the minimal transitive dissimilarities as defined in Equation (5). The transitive dissimilarities are equal to the edge weights.*

PROOF. We prove Proposition 2 using dynamic programming. Start from two-hop paths V_i-V_k-V_j between any given vertices V_i and V_j. As the edge weights d satisfy the minimal transitive dissimilarities, so d_{ij} must be less than or equal to two-hop transitive weight $T(P_{ikj})$ for any k. Since we have minimal transitive dissimilarity $m_{ij} \leq d_{ij}$ implied by Equation (5), so $m_{ij} \leq d_{ij} \leq T(P_{ikj})$ holds. For two-hop minimal transitive dissimilarity, we get $m_{ij} = d_{ij}$.

Given any three-hop path between V_i and V_j, denoted as V_i-V_k-V_l-V_j, we can change V_i-V_k-V_l to V_i-V_l, or change V_k-V_l-V_j to V_k-V_j based on the destination from two-hop paths. We apply transitive dissimilarity and the edge weight equivalence property again on path V_i-V_l-V_j or V_i-V_k-V_j again; then, we get $m_{ij} = d_{ij}$, for any path V_i-V_k-V_l-V_j.

For any n-hop path ($n \geq 2$), the same process can be applied. Thus, Proposition 2 is proved. □

PROPOSITION 3. *Given node pair V_i and V_j, let V_i-V_{k1}-$\cdots$-V_{km}-V_j) be the path with the eventual minimal transitive dissimilarity. After successive tightening of edges V_i-V_{k1}, V_{k1}-V_{k2}, $\ldots$, V_{km}-V_j in order, the transitive dissimilarity achieves the final optimal minimal transitive dissimilarity. This holds no matter what other edge relaxations occur.*

PROOF. Since the eventual path between V_i and V_j with minimal transitive dissimilarity is given, the length-2 minimal transitive dissimilarity (optimal solution) can be easily obtained. Also, the length-3 minimal transitive dissimilarity can be obtained based on the length-2 solution, and it is obviously the optimal solution. The conclusion holds when extending to the last edge of the path. Thus, Proposition 3 is proved. □

PROPOSITION 4. *Algorithm 1 correctly computes the minimum transitive dissimilarity.*

PROOF. The outer loop $k = 1$ to N guarantees that all paths between any given vertices V_i and V_j will be considered to achieve the eventual optimal path. Proposition 3 ensures that the final correct solution will be reached no matter how internal vertices along the path are involved. Proposition 2 guarantees that any optimal solution obtained before traversing all the possible solutions will be maintained without change in the future. □

From these propositions, we know that the minimal transitive dissimilarity brings objects closer than the original distance matrix. Our experimental results in Section 8 show that the final hierarchical solutions arrived at by fitting an ultra-meric using transitive dissimilarity generally outperform the method that directly performs SL and CL hierarchical clusterings on the aggregated descriptor matrices. A formal analysis of cluster separation enhancement requires dedicated work and is one of our future projects.

7. DENDROGRAM SELECTION

Selecting a subset of input clusterings to form a smaller ensemble has been shown to achieve better performance than using all available solutions for partitional clustering methods [Azimi and Fern 2009; Fern and Lin 2008]. The selection is based on the quality and diversity of each individual clustering solution. For partitional clustering, since the clustering solutions are naturally represented using vectors or matrices [Li and Ding 2008; Strehl and Ghosh 2003], the diversity and quality of the clustering solutions can be easily computed. To perform dendrogram selection, the question is how to compute the diversity and quality of different hierarchical clustering solutions.

We propose two approaches to perform dendrogram selection based on tree distances. First, we introduce the tree distances to measure the similarities/differences between different hierarchies. Two distances are frequently used in the literature to compute the distance between two evolutionary dendrograms: Branch Score Distance (BSD) of Kuhner and Felsenstein [1994] and Symmetric Difference (SD) of Robinson and Foulds [1981]. Both distances are computed by considering all possible branches that could exist on the two trees. Note that each branch makes a partition of the given dataset into two groups—the ones connected to one end of the branch (the ones on a subtree) and the ones connected to the other (the others). BSD uses branch lengths, whereas SD does not use branch lengths and only uses the tree topologies. For BSD, each partition on a dendrogram has an associated branch length (i.e., the distance when merging two subclusters). BSD is then computed by taking the sum of squared differences between the branch lengths of two dendrograms. SD is calculated as the number of partitions that only exist in one of the dendrograms.

The goal of dendrogram selection is to select a diverse subset of dendrograms where each of them has good quality. We propose two approaches for dendrogram selection using tree distances. In both approaches, the size of the selected set of dendorgrams is given as an input. The first approach is to use a modified K-medoids algorithm (with the tree distances) to cluster those dendrograms and then select the medoids for each cluster. The medoid of a cluster is a representative object whose average similarity to all the other objects in the cluster is maximized; thus, the medoid dendrogram can be considered to best capture the information contained in the cluster and has good quality. On the other hand, selecting medoids from different clusters achieves diversity.

The second approach is based on the farthest-point heuristic [Gonzalez 1985]. The approach starts with the medoid of all the input clustering solutions. Then, pick a dendrogram that is as far from the selected dendrogram as possible. In general, the approach picks a dendrogram to maximize the distances to the nearest of all

Table II. Dataset Descriptions

Name	# Samples	# Attributes	# Classes
Wine	178	13	3
Parkinson Disease	195	22	2
Libras Movement	360	90	15
WebACE	2340	1000	12
Reuters	2787	1000	9

Table III. Experimental Results on Wine Dataset Using All Input Dendrograms

Descriptor	Ultra	Single-Link
CD	**0.392**	0.381
CMD	**0.443**	0.273
MED	**0.292**	0.288
PMD	**0.267**	0.232
SMD	**0.299**	0.290

The maximum CPCC value for any input dendrogram is 0.407, and the average value of all input dendrograms is 0.282.

dendrograms picked so far. Specifically, if $t_1, t_2, \ldots, t_{i-1}$ denote the selected dendrograms so far, then we pick t_i to maximize

$$min\{dist(s_i, s_1), dist(s_i, s_2), \ldots, dist(s_i, s_{i-1})\}. \tag{7}$$

The approach stops after the required number of dendrogram has been selected.

8. EXPERIMENTS

8.1. Experiment Setup

To evaluate our proposed ensemble framework, we focus on how well the ensemble hierarchical solution reflects the characteristics of the original dataset. **Co-Phenetic Correlation Co-efficiency (CPCC)** is used as the performance measure [Rohlf and Fisher 1968; Sokal and Rohlf 1962]. It aims to evaluate how faithfully a dendrogram preserves the pair-wise distances between the original data samples, and it can be calculated as

$$c = \frac{\sum_{i<j}(d(i,j) - d)(h(i,j) - h)}{\sqrt{[\sum_{i<j}(d(i,j) - d)^2][\sum_{i<j}(h(i,j) - h)^2]}}, \tag{8}$$

where $d(i, j)$ is the distance between the i-th and j-th data instances, $h(i,j)$ is the height of lowest common ancestor of the i-th and j-th data instances in ensemble dendrogram, d is the averages of $d(i, j)$ over all pairs, and h is the average of $h(i, j)$. Generally, the higher the CPCC value, the better the clustering performance.

We use five datasets from different domains to conduct the experiments: three datasets (Wine, Parkinson Disease, and Libras Movement) from UCI Machine Learning Repository,[1] and two benchmark text datasets for document clustering (WebACE and Reuters datasets) [Li and Ding 2008]. The datasets and their characteristics are summarized in Table II. The two text datasets are represented using the vector space model, and they are also preprocessed by removing the stop words and unnecessary tags and headers. All experiments are conducted under the environment of Windows XP operating system plus Intel P4 1.83GHz CPU and 4GB of RAM.

[1]The datasets can be downloaded from http://archive.ics.uci.edu/ml/.

Table IV. Experimental Results on Parkinson Disease Dataset Using All Input Dendrograms

Descriptor	Ultra	Single-Link
CD	**0.577**	0.554
CMD	**0.431**	0.419
MED	**0.485**	0.428
PMD	0.402	**0.417**
SMD	0.448	**0.491**

The maximum CPCC value for any input dendrogram is 0.381 and the average value of all input dendrograms is 0.201.

Table V. Experimental Results on Libra Movement Dataset Using All Input Dendrograms

Descriptor	Ultra	Single-Link
CD	**0.423**	0.419
CMD	**0.411**	0.389
MED	0.36	**0.363**
PMD	**0.279**	0.266
SMD	**0.45**	0.438

The maximum CPCC value for any input dendrogram is 0.334 and the average value of all input dendrograms is 0.25.

Table VI. Experimental Results on WebACE Dataset Using All Input Dendrograms

Descriptor	Ultra	Complete-Link
CD	**0.465**	0.4637
CMD	**0.4971**	0.4963
MED	**0.4787**	0.4699
PMD	0.4831	**0.4896**
SMD	**0.5056**	0.4781

The maximum CPCC value for any input dendrogram is 0.47 and the average value of all input dendrograms is 0.428.

8.2. Ensemble Hierarchical Clusterings

In this set of experiments, for each dataset, 10 input dendrograms are generated by using different hierarchical clustering methods on different attribute subsets. In particular, they are generated as follows: (1) five different attribute subsets are randomly constructed first, each of which contains 90% of all the attributes; and (2) SL and CL algorithms are applied to different attribute subsets.

We evaluate our proposed method for generating the final hierarchical solution by fitting an ultra-meric using all five dendrogram descriptors (i.e., CD, CMD, MED, PMD, SMD). We also compare our proposed method (denoted as *ultra* in the experimental results) with the method that directly performs SL and CL hierarchical clusterings on the aggregated descriptor matrices (denoted as *single-link*/*complete-link* or *SL*/*CL*.[2]

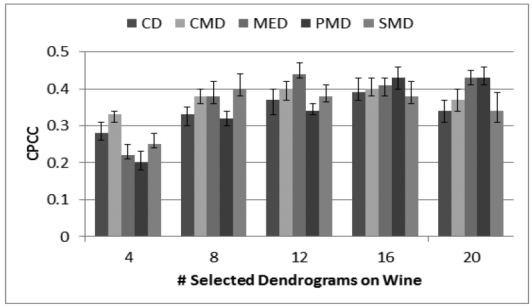
8.2.1. Results Using All Input Dendrograms. Tables III–VII present the experimental results on **six** datasets using all input dendrograms, respectively. Note that, unlike ensemble clustering for partitional clustering results, for hierarchical clustering ensembles, once the set of individual hierarchical clustering results is fixed, then the result of the ensemble is also determined. From the experimental results, we observe that:

[2]In our work, we apply *SL* on the aggregated descriptor matrices for four UCI datasets and apply *CL* on the aggregated descriptor matrices for two text datasets.

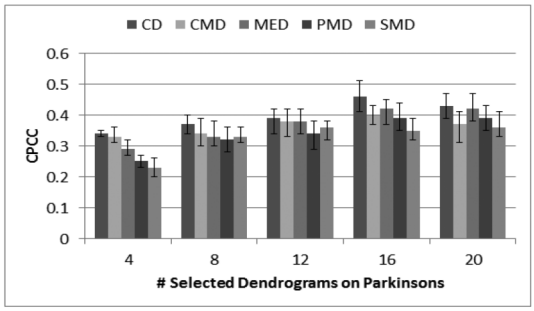
Table VII. Experimental Results on Reuters Dataset Using All Input Dendrograms

Descriptor	Ultra	Complete-Link
CD	**0.7349**	0.7312
CMD	**0.7822**	0.7435
MED	**0.7415**	0.7176
PMD	**0.7624**	0.6955
SMD	0.6475	**0.6479**

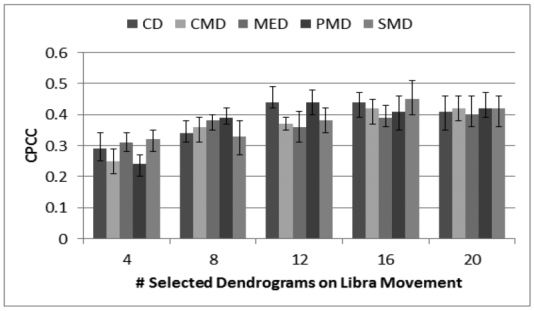
The maximum CPCC value for any input dendrogram is 0.7583 and the average value of all input dendrograms is 0.633.



(a) Wine

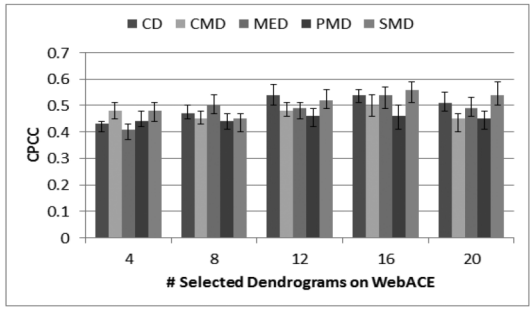


(b) Parkinsons

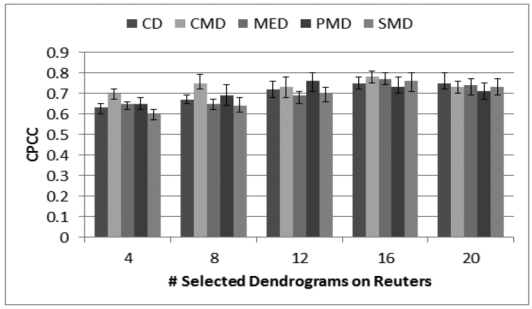


(c) Libra Movement

Fig. 4. The performance variation on different numbers of selected dendrograms over 20 trials.



(a) WebACE



(b) Reuters

Fig. 5. The performance variation on different numbers of selected dendrograms over 20 trials.

(1) Our proposed method *ultra* generally outperforms hierarchical clustering (*SL* or *CL*) across various descriptors on most counts, especially on large datasets (e.g., WebACE and Reuters), and (2) the ensemble solution using all input dendrograms may be worse than the best individual dendrogram, thus demonstrating the need for ensemble selection.

8.2.2. Results on Different Input Dendrograms. In order to provide more insights into our proposed method, we also conduct experiments with different sets of input dendrograms. Figures 4 and 5 show the experimental results on the three UCI datasets (Wine, Parkinsons, and Libra Movement) and the two text datasets (WebACE and Reuters), respectively, with different sets of input dendrograms. In particular, for a given size, we randomly select a set of input dendrograms, and then perform the experiments. The reported results are averaged over 20 different runs.

Based on our observation, the best performance is often obtained when the number of input dendrograms is around 4 or 5. Although this experiment is conducted by randomly selecting input dendrograms, it still demonstrates that using a subset of input dendrograms (rather than using all dendrograms) may improve the ensemble performance. The issue of using dendrogram selection strategies to form the candidate subset is discussed in Section 8.2.3 and Section 8.2.4, respectively.

8.2.3. Experiments on Ensemble Selection. We also conducted experiments to demonstrate the effects of ensemble selection. Note that dendrogram selection can be performed using two different approaches (K-medoid and Farthest neighbor, denoted as K and F) with two different distances (Branch Length Score Distance or Symmetric Distance, denoted as B and S). Tables VIII–Table XII present the experimental results

Table VIII. Experimental Results on Wine Dataset Using Six Selected Input Dendrograms

Desc.	Sel	Dis	max	ave	ultra	SL
CD	F	B	0.292	0.245	**0.352**	0.331
	K	B	0.306	0.251	**0.373**	0.357
	F	S	0.281	0.229	**0.329**	0.292
	K	S	0.299	0.238	0.336	**0.344**
CMD	F	B	0.292	0.245	**0.387**	0.378
	K	B	0.306	0.251	**0.373**	0.365
	F	S	0.281	0.229	**0.361**	0.329
	K	S	0.299	0.238	**0.35**	0.337
MED	F	B	0.292	0.245	**0.369**	0.348
	K	B	0.306	0.251	**0.355**	0.316
	F	S	0.281	0.229	**0.339**	0.318
	K	S	0.299	0.238	**0.357**	0.323
PMD	F	B	0.292	0.245	**0.296**	0.284
	K	B	0.306	0.251	0.315	**0.331**
	F	S	0.281	0.229	**0.316**	0.302
	K	S	0.299	0.238	0.305	**0.32**
SMD	F	B	0.292	0.245	**0.321**	0.307
	K	B	0.306	0.251	**0.338**	0.32
	F	S	0.281	0.229	**0.317**	0.293
	K	S	0.299	0.238	**0.309**	0.304

K and F denote K-medoid and Farthest Neighbor of ensemble selection methods, respectively, and B and S denote Branch Length Score Distance and Symmetric Distance of dendrogram distances, respectively.

Table IX. Experimental Results on Parkinson Disease Dataset Using Four Selected Input Dendrograms

Desc.	Sel	Dis	max	ave	ultra	SL
CD	F	B	0.438	0.256	**0.549**	0.521
	K	B	0.467	0.251	0.538	**0.544**
	F	S	0.493	0.273	**0.537**	0.505
	K	S	0.452	0.235	**0.526**	0.524
CMD	F	B	0.438	0.256	**0.56**	0.512
	K	B	0.467	0.251	**0.572**	0.542
	F	S	0.493	0.273	**0.553**	0.527
	K	S	0.452	0.235	0.524	**0.536**
MED	F	B	0.438	0.256	**0.574**	0.539
	K	B	0.467	0.251	**0.595**	0.532
	F	S	0.493	0.273	**0.54**	0.537
	K	S	0.452	0.235	**0.589**	0.527
PMD	F	B	0.438	0.256	**0.517**	0.492
	K	B	0.467	0.251	0.523	**0.531**
	F	S	0.493	0.273	**0.502**	0.499
	K	S	0.452	0.235	**0.544**	0.507
SMD	F	B	0.438	0.256	0.529	0.529
	K	B	0.467	0.251	**0.551**	0.504
	F	S	0.493	0.273	**0.547**	0.516
	K	S	0.452	0.235	0.498	**0.511**

K and F denote K-medoid and Farthest Neighbor of ensemble selection methods, respectively, and B and S denote Branch Length Score Distance and Symmetric Distance of dendrogram distances, respectively.

Table X. Experimental Results on Libra Movement Dataset Using Four Selected Input Dendrograms

Desc.	Sel	Dis	max	ave	ultra	SL
CD	F	B	0.287	0.199	0.392	**0.433**
	K	B	0.291	0.185	**0.441**	0.408
	F	S	0.274	0.167	**0.4**	0.396
	K	S	0.303	0.158	**0.398**	0.385
CMD	F	B	0.287	0.199	**0.432**	0.424
	K	B	0.291	0.185	**0.446**	0.418
	F	S	0.274	0.167	**0.410**	0.402
	K	S	0.303	0.158	**0.453**	0.391
MED	F	B	0.287	0.199	**0.49**	0.458
	K	B	0.291	0.185	0.442	**0.476**
	F	S	0.274	0.167	**0.483**	0.472
	K	S	0.303	0.158	0.453	**0.461**
PMD	F	B	0.287	0.199	**0.397**	0.346
	K	B	0.291	0.185	**0.383**	0.315
	F	S	0.274	0.167	**0.401**	0.359
	K	S	0.303	0.158	**0.394**	0.329
SMD	F	B	0.287	0.199	**0.437**	0.384
	K	B	0.291	0.185	**0.462**	0.391
	F	S	0.274	0.167	0.423	**0.439**
	K	S	0.303	0.158	**0.468**	0.379

K and F denote K-medoid and Farthest Neighbor of ensemble selection methods, respectively, and B and S denote Branch Length Score Distance and Symmetric Distance of dendrogram distances, respectively.

Table XI. Experimental Results on WebACE Dataset Using Four Selected Input Dendrograms

Desc.	Sel	Dis	max	ave	ultra	CL
CD	F	B	0.483	0.41	**0.491**	0.49
	K	B	0.474	0.409	**0.505**	0.499
	F	S	0.465	0.417	0.492	0.492
	K	S	0.487	0.405	**0.501**	0.494
CMD	F	B	0.483	0.41	**0.511**	0.501
	K	B	0.474	0.409	**0.509**	0.507
	F	S	0.465	0.417	0.498	**0.503**
	K	S	0.487	0.405	**0.505**	0.497
MED	F	B	0.483	0.41	**0.513**	0.502
	K	B	0.474	0.409	**0.504**	0.497
	F	S	0.465	0.417	**0.5**	0.497
	K	S	0.487	0.405	**0.507**	0.489
PMD	F	B	0.483	0.41	0.496	**0.498**
	K	B	0.474	0.409	0.492	**0.497**
	F	S	0.465	0.417	**0.501**	0.5
	K	S	0.487	0.405	**0.498**	0.49
SMD	F	B	0.483	0.41	**0.503**	0.491
	K	B	0.474	0.409	**0.5**	0.493
	F	S	0.465	0.417	**0.499**	0.484
	K	S	0.487	0.405	**0.507**	0.495

K and F denote K-medoid and Farthest Neighbor of ensemble selection methods, respectively, and B and S denote Branch Length Score Distance and Symmetric Distance of dendrogram distances, respectively.

Table XII. Experimental Results on Reuters Dataset Using Four Selected Input Dendrograms

Desc.	Sel	Dis	max	ave	ultra	CL
CD	F	B	0.73	0.682	**0.747**	0.739
	K	B	0.741	0.635	0.785	**0.794**
	F	S	0.737	0.696	**0.792**	0.786
	K	S	0.729	0.64	**0.769**	0.75
CMD	F	B	0.73	0.682	**0.793**	0.767
	K	B	0.741	0.635	**0.798**	0.752
	F	S	0.737	0.696	**0.794**	0.755
	K	S	0.729	0.64	**0.782**	0.751
MED	F	B	0.73	0.682	**0.779**	0.754
	K	B	0.741	0.635	**0.783**	0.781
	F	S	0.737	0.696	0.765	**0.77**
	K	S	0.729	0.64	**0.752**	0.75
PMD	F	B	0.73	0.682	**0.782**	0.763
	K	B	0.741	0.635	**0.775**	0.755
	F	S	0.737	0.696	**0.787**	0.761
	K	S	0.729	0.64	0.74	**0.745**
SMD	F	B	0.742	0.726	**0.797**	0.784
	K	B	0.744	0.727	**0.782**	0.753
	F	S	0.736	0.730	**0.771**	0.767
	K	S	0.731	0.722	0.75	0.75

K and F denote K-medoid and Farthest Neighbor of ensemble selection methods, respectively, and B and S denote Branch Length Score Distance and Symmetric Distance of dendrogram distances, respectively.

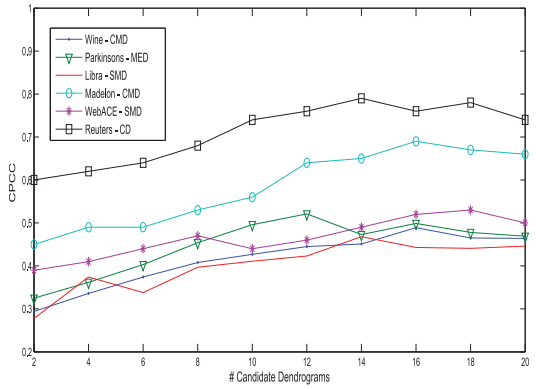
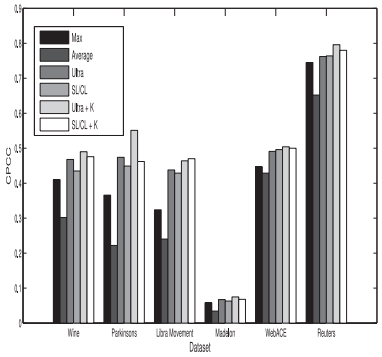


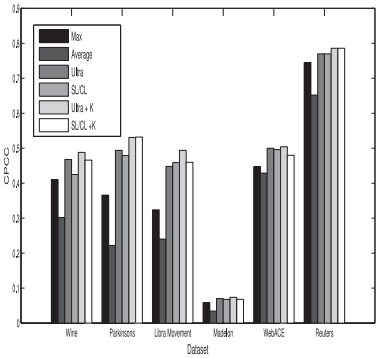
Fig. 6. The performance variation on all datasets with different numbers of candidate dendrograms.

on the **six** datasets using four selected input dendrograms, respectively.[3] In these tables, *Sel* denotes the ensemble selection approaches, *Dis* represents the tree distances, *max* represents the maximum CPCC value for any input dendrogram, and ave represents the average CPCC value for the input dendrograms. The experiments show that: (1) with ensemble selection, the results of both *ultra* and hierarchical clustering (*SL* or *CL*) have improved; (2) *ultra* still outperforms hierarchical clustering (*SL* or *CL*) in most cases; and (3) in many cases, the experiment results of *ultra* and hierarchical

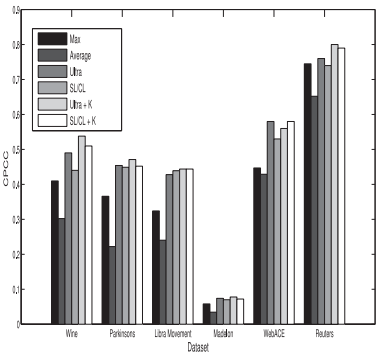
[3]The value of 4 is chosen based on our experiments on ensemble size selection, and it seems to provide good results in our experiments. How to come up with a principled way to determine ensemble size selection is one of our future projects.



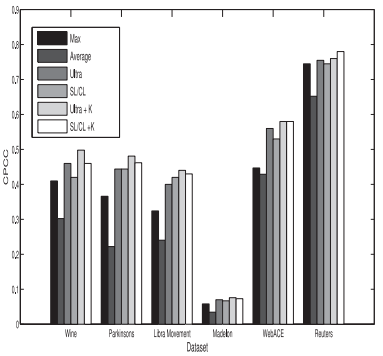
(a) The five dendrograms are represented by Cophenetic Distance Matrix (CD) and are selected using Farthest Neighbor ensemble selection and Branch Score Distance.



(b) The five dendrograms are represented by Cophenetic Distance Matrix (CD) and are selected using K-Medoid ensemble selection and Symmetric Distance.



(c) The five dendrograms are represented by Cluster Membership Divergence (CMD) and are selected using K-Medoid ensemble selection and Branch Score Distance.



(d) The five dendrograms are represented by Cluster Membership Divergence (CMD) and are selected using Farthest Neighbor ensemble selection and Symmetric Distance.

Fig. 7. The performance comparison of combining 10 partitional clustering results with five selected dendrograms. *max* represents the maximum CPCC value for any input dendrogram, and ave represents the average CPCC value for the input dendrograms. *ultra* and *SL*/*CL* represent the recovery approaches for ensemble dendrograms by using ultra-matrix transformation and hierarchical clustering, respectively. *ultra+K* and *SL*/*CL+K* represent the combination of K-means clustering results and the previous two methods.

clustering (*SL* or *CL*) outperform the best dendrogram in the candidate set, which means those ensemble dendrograms could be more representative of the original set.

8.2.4. Experiments on Ensemble Size. To demonstrate the effect of the size of the ensemble, Figures 4 and 5 show the performance variation on different numbers of selected dendrograms on all datasets. We apply K-Medoid selection methods on SD to choose candidate dendrograms. For each dataset, we vary the group size of candidate dendrograms and use CMD as the descriptor to conduct the dendrogram selection.

Figure 6 shows the CPCC value for each dendrogram group, averaging over 20 runs. Note that for better readability, the plotted value of the Madelon dataset is 10 times its actual value. The performance slightly decreases once the number of ensemble dendrograms reaches a certain size. So selecting a relatively smaller subset is likely to

produce better ensemble results. It also shows that ensemble selection can influence the ensemble results and can be used to produce better hierarchical solutions.

8.3. Ensemble Partitional and Hierarchical Clusterings

In this set of experiments, we evaluate our proposed method for combining both partitional and hierarchical clusterings on all datasets. For each dataset, 10 partitional clustering results are obtained by running K-means 10 times, and they are combined with five input dendrograms. Figure 7 presents the experimental results. From the experimental results, we conclude that our ensemble framework is able to combine both partitional and hierarchical clusterings and improve the performance on most datasets. The results also show that our proposed method *ultra* clearly outperforms *SL*/*CL* on all datasets, and *ultra+K* generally outperforms *SL*/*CL+K* in most cases.

9. CONCLUSION AND FUTURE WORK

A framework for ensemble hierarchical clusterings based on descriptor matrices is proposed in this article. Three important components of the framework (dendrogram selection, dendrogram description, and dendrogram combination) are studied. In particular, two ensemble selection schemes based on tree distances are proposed, five different dendrogram descriptor matrices are investigated, and a novel method for fitting an ultra-metric from the aggregated descriptor matrix is developed. Since partitional clustering results can be easily represented using distance matrices, our descriptor matrices-based framework can be naturally generalized to ensemble both partitional clustering and hierarchical clustering results as partitional clustering results. Experiments are conducted to demonstrate the effectiveness of our proposed approaches.

There are several avenues for future work. First, we plan to investigate the techniques for scaling up the ensemble process to large-scale datasets. Second, our studies show that selecting a relatively smaller subset is likely to produce better ensemble results. One interesting question is how to determine the ensemble size. Another interesting yet related direction is that, rather than picking representative dendrograms, we can associate every generated dendrogram with a weight. So, when considering the ensemble, dendrograms with larger weights can contribute more than can dendrograms with smaller weights. Third, another aspect of interest is to provide a formal analysis on cluster separation enhancement using transitive dissimilarity.

REFERENCES

E. N. Adams. 1986. N-trees as nestings: Complexity, similarity, and consensus. *Journal of Classification* 3, 299–317. 10.1007/BF01894192.

E. N. Adams III. 1972. Consensus techniques and the comparison of taxonomic trees. *Systematic Zoology* 21, 4, 390–397.

R. Agarwala, V. Bafna, M. Farach, M. Paterson, and M. Thorup. 1999. On the approximability of numerical taxonomy (fitting distances by tree metrics). *SIAM Journal on Computing* 1073–1085.

N. Ailon and M. Charikar. 2005. Fitting tree metrics: Hierarchical clustering and phylogeny. In *Proceedings of the Symposium on Foundations of Computer Science*. 73–82.

J. Azimi and X. Fern. 2009. Adaptive cluster ensemble selection. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'09)*. 992–997.

L. Breiman and L. Breiman. 1996. Bagging predictors. *Machine Learning* 24, 2, 123–140, Aug. 1996.

L. Breiman and L. Breiman. 2001. Random forests. *Machine Learning* 5–32.

C. Ding, X. He, H. Xiong, H. Peng, and S. R. Holbrook. 2006. Transitive closure and metric inequality of weighted graphs: Vdetecting protein interaction modules using cliques. *International Journal of Data Mining and Bioinformatics* 1, 162–177.

M. Farach, T. M. Przytycka, and M. Thorup. 1995. On the agreement of many trees. *Information Processing Letter* 55, 297–301.

X. Z. Fern and C. E. Brodley. 2004. Solving cluster ensemble problems by bipartite graph partitioning. In *Proceedings of the 21st International Conference on Machine Learning (ICML'04)*. ACM, New York, NY, 36.

X. Z. Fern and W. Lin. 2008. Cluster ensemble selection. *Statistical Analysis and Data Mining* 1, 128–141.

C. Fraley and A. E. Raftery. 1998. How many clusters? Which clustering method? Answers via model-based cluster analysis. *The Computer Journal* 41, 578–588, 1998.

A. Gionis, H. Mannila, and P. Tsaparas. 2005. Clustering aggregation. In *Proceedings of the 21st International Conference on Data Engineering (ICDE'05)*. 341–352.

T. Gonzalez. 1985. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38, 293–306, 1985.

M. Hossain, S. M. Bridges, Y. Wang, and J. E. Hodges. 2012. An effective ensemble method for hierarchical clustering. In *Proceedings of the 5th International C* Conference on Computer Science and Software Engineering*. ACM, 18–26.

A. Jain and R. Dubes. 1998. *Algorithms for Clustering Data*. Prentice Hall advanced reference series. Prentice Hall, 1988.

M. Jalalat-evakilkandi and A. Mirzaei. 2010. A new hierarchical-clustering combination scheme based on scatter matrices and nearest neighbor criterion. In *Proceedings of the 2010 5th International Symposium on Telecommunications (IST'10)*. IEEE, 904–908.

K. Koutroumbas, I. Tsagouri, and A. Belehaki. 2010. On the clustering of foF2 time series corresponding to disturbed ionospheric periods. *Advances in Space Research* 45, 9, 1129–1144.

M. K. Kuhner and J. Felsenstein. 1994. A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. *Molecular Biology and Evolution* 11, 3, 459–68.

T. Li and C. Ding. 2008. Weighted consensus clustering. In *Proceedings of the SIAM International Conference on Data Mining*. 798–809.

T. Li, C. Ding, and M. I. Jordan. 2007. Solving consensus and semi-supervised clustering problems using nonnegative matrix factorization. In *Proceedings of the 2007 7th IEEE International Conference on Data Mining (ICDM'07)*. IEEE Computer Society, Washington, DC, 577–582.

T. Li, M. Ogihara, and S. Ma. 2004. On combining multiple clusterings. In *Proceedings of the 13th ACM International Conference on Information and Knowledge Management (CIKM'04)*. ACM, New York, NY, 294–303.

T. Li, M. Ogihara, and S. Ma. 2010. On combining multiple clusterings: an overview and a new perspective. *Applied Intelligence* 33, 2, 207–219.

Y. Lu and Y. Wan. 2012. PHA: A fast potential-based hierarchical agglomerative clustering method. *Pattern Recognition* 46, 5, 1227–1239, May 2013.

D. Luo, C. Ding, H. Huang, and F. Nie. 2011. Consensus spectral clustering in near-linear time. In *Proceedings of the 2011 IEEE 27th International Conference on Data Engineering (ICDE'11)*. IEEE Computer Society, Washington, DC, 1079–1090.

H. D. Meyer, H. Naessens, and B. D. Baets. 2004. Algorithms for computing the min-transitive closure and associated partition dendrogram of a symmetric fuzzy relation. *European Journal of Operational Research* 155, 1, 226–238.

A. Mirzaei and M. Rahmati. 2008. Combining hierarchical clusterings using min-transitive closure. In *Proceedings of the 19th International Conference on Pattern Recognition (ICPR'08)*. IEEE, 1–4.

A. Mirzaei and M. Rahmati. 2010. A novel hierarchical-clustering-combination scheme based on fuzzy-similarity relations. *IEEE Transactions on Fuzzy Systems* 18, 1, 27–39.

A. Mirzaei, M. Rahmati, and M. Ahmadi. 2008. A new method for hierarchical clustering combination. *Intelligent Data Analysis* 12, 549–571.

S. Monti, P. Tamayo, J. Mesirov, and T. Golub. 2003. Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data. *Machine Learning* 52, 91–118.

J. Podani. 2000. Simulation of random dendrograms and comparison tests: Some comments. *Journal of Classification* 17, 123–142.

E. Rashedi and A. Mirzaei. 2011. A novel multi-clustering method for hierarchical clusterings based on boosting. In *Proceedings of the 2011 19th Iranian Conference on Electrical Engineering (ICEE'11)*. IEEE, 1–4.

D. F. Robinson and L. R. Foulds. 1981. Comparison of phylogenetic trees. *Mathematical Bioscience*, 53, 131–147.

F. J. Rohlf and D. R. Fisher. 1968. Tests for hierarchical structure in random data sets. *Systematic Zoology* 17, 4, 407–412.

R. E. Schapire and Y. Singer. 1999. Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, 37, 297–336, 1999.

R. R. Sokal and F. J. Rohlf. 1962. The comparison of dendrograms by objective methods. *Taxon*, 11, 2, 1962.

A. Strehl and J. Ghosh. 2003. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3, 583–617, March 2003.

C. A. Sugar and G. M. James. 2003. Finding the number of clusters in a data set: An information theoretic approach. *Journal of the American Statistical Association* 98, 750–763, 2003.

D. Swofford. 1991. When are phylogeny estimates from molecular and morphological data incongruent? In M. M. Miyamoto and J. Cracraft, editors, *Phylogenetic Analysis of DNA Sequences*. Oxford University Press, 295–333.

P.-N. Tan, M. Steinbach, and V. Kumar. 2005. *Introduction to Data Mining* (1st ed.). Addison-Wesley Longman, Boston, MA.

R. Tibshirani, G. Walther, and T. Hastie. 2001. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society Series B* 63, 2, 411–423.

A. Topchy, A. Jain, and W. Punch. 2005. Clustering ensembles: models of consensus and weak partitions. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27, 12, 1866–1881.

M. Wilkinson. 1994. Common cladistic information and its consensus representation: Reduced adams and reduced cladistic consensus trees and profiles. *Systematic Biology*, 43, 3, 343–368, 1994.

D. H. Wolpert. 1992. Stacked generalization. *Neural Networks*, 5, 241–259, 1992.

J. Wu, H. Xiong, and J. Chen. 2009. Towards understanding hierarchical clustering: A data distribution perspective. *Neurocomputing*, 72, 10–12, 2319–2330, 2009.

Y. Zhao and G. Karypis. 2002. Evaluation of hierarchical clustering algorithms for document datasets. In *Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM'02)*. ACM, New York, NY, 515–524.

L. Zheng and T. Li. 2011. Semi-supervised hierarchical clustering. In *Proceedings of the 2011 IEEE 11th International Conference on Data Mining (ICDM'11)*, 982–991, 2011.

L. Zheng, T. Li, and C. H. Q. Ding. 2010. Hierarchical ensemble clustering. In *ICDM'10*, 1199–1204, 2010.

Received July 2013; revised January 2014; accepted March 2014

An Empirical Study of Ontology-Based Multi-Document Summarization in Disaster Management

Lei Li and Tao Li

***Abstract*—Domain ontology, as a conceptual model, provides a meaningful framework for semantic representation of textual information. In this paper, we explore the feasibility of using the ontology in solving multi-document summarization problems in the domain of disaster management. We provide an empirical study of different approaches in which the ontology has been used for summarization tasks. Extensive experiments on a collection of press releases relevant to Hurricane Wilma in 2005 demonstrate that ontology-based multi-document summarization methods outperform other baselines in terms of the summary quality.**

***Index Terms*—Disaster management, multi-document summarization, ontology, query expansion.**

I. Introduction

IT IS WELL KNOWN that hurricanes, earthquakes, and other natural disasters cause immense physical destruction and loss of life and property around the world. In order to efficiently analyze the trend of the disasters and minimize the consequent loss for future situation, effective information gathering methods are important. Specifically, a myriad of news and reports that are related to the disaster may be recorded in the form of text documents. The domain experts expect to obtain condensed information about the detailed disaster event description, e.g., the evolutionary tendency of the disaster, the operational status of the public services, and the reconstruction process of the homestead. In the following, a representative scenario is provided, in which the information frequently investigated by a disaster analyst is described.

Scenario: Hurricane Wilma passed through South Florida in October, 2005. During Wilma, the power supply in Miami was extremely influenced. The domain experts want to check the status of the power supply during Wilma and after Wilma passed.

Manuscript received March 18, 2011; revised August 12, 2011, February 3, 2012, and July 7, 2012; accepted April 2, 2013. Date of publication November 13, 2013; date of current version January 13, 2014. This work was supported in part by the National Science Foundation under Grant HRD-0833093, Grant CNS-1126619, and Grant IIS-1213026, and the U.S. Department of Homeland Security under grant Award Number 2010-ST-062-000039, and Army Research Office under Grant W911NF-10-1-0366 and Grant W911NF-12-1-0431. This paper was recommended by Associate Editor Z. Zdrahal.

The authors are with the School of Computing and Information Sciences, Florida International University, Miami, FL 33199 USA (e-mail: lli003@cs.fiu.edu; taoli@cs.fiu.edu).

Digital Object Identifier 10.1109/TSMCC.2013.2258335

TABLE I
EXAMPLE OF DISASTER INFORMATION

Power supply in Miami
Florida Power and Light reports 380 000 customers have lost power on October 21th, 2005.
Nearly one million FPL customers without power in Miami-Dade County on October 24th, 2005.
Power is beginning to be restored to FPL customers from October 25th, and it may take several weeks to be fully restored.

A list of descriptive sentences on this topic are shown in Table I. As is shown, the three sentences provide a summary on the status of power supply over a week in the district of Miami-Dade County. Such information can provide domain analysts a preliminary overview of how the power supply was influenced by the hurricane, and subsequently, domain analysts will contact the corresponding department and establish a set of measures that would be helpful once the situation happens again.

In the domain of disaster management, over thousands of hundreds of reports are often released by the local government or local emergency offices during the disaster, which cover most events relevant to the disaster and the time span will be days to months, depending on how severe the disaster is. The data will be presented in a format of newswire, containing a lot of routine reporting on multiple aspects of the disaster. In such a case, it is extremely difficult for domain experts to quickly find either the most important information overall (generic summarization) or the most relevant information to a specified query (query/topic-focused summarization). Therefore, multi-document summarization techniques can be used to extract meaningful information from multiple reports.

A domain ontology related to disaster management, describing the concepts and the corresponding relations of these concepts, is often provided by domain experts [1]. Such an ontology contains plentiful conceptual information related to the document set, which may be beneficial for users to summarize the documents. A natural question is how we can utilize the ontology to obtain high-quality summaries, i.e., representing topics with nonredundant sentences.

In this paper, we explore the feasibility of employing the ontology into multi-document summarization problems in disaster management domain. We first discuss how to represent

a sentence as a vector using the domain ontology. We then delve into the problems from two directions: generic and query-focused summarization. In generic summarization, we provide comprehensive studies of the centroid-based sentence selection approaches by using different vector space models, and explore the possibility of utilizing the ontology to achieve the goal of reducing information redundancy. In query-focused summarization, we optimize the final summary results by employing ontology-based query expansion methods into the summarization. We conduct experiments on a collection of press releases related to Hurricane Wilma, and the results show that ontology-based methods can provide promising performance for summarization.

The rest of the paper is organized as follows. In Section II, we review the related work. After introducing the domain ontology applied in our work in Section III, we provide a comparative study on several ontology-based representations in Section IV. Section V presents the experimental results and analysis, and finally Section VI concludes the paper and covers the future work.

II. Related Work

A. Generic Summarization

For generic summarization, a saliency score is usually assigned to each sentence, the sentences are ranked according to the saliency score, and then the top ranked sentences are selected as the summary based on the ranking result. Recently, both unsupervised and supervised methods have been proposed to analyze the information contained in a document set, and extract highly salient sentences into the summary based on syntactic or statistical features [2]–[6]. For example, MEAD [7] is an implementation of the centroid-based method in which the sentence scores are computed based on sentence-level and inter-sentence features.

However, most existing methods ignore the conceptual information in the sentence level. In most cases, the conceptual information can provide users more readable results for summaries. Some researchers utilize the explicit concepts within sentences to address multi-document summarization [8], [9], e.g., using Wikipedia. However, such techniques cannot be directly applied to domain-specific document summarization tasks, since Wikipedia contains too many concepts not relevant to a specific domain. In our previous work [25], we explored the possibility of using domain-specific ontology for multi-document summarization; however, no detailed semantic relationship is considered.

B. Query-Focused Summarization

In query-focused summarization, the information related to a given topic or query should be incorporated into summaries, and the sentences suiting the user's declared information need should be extracted. Many methods for generic summarization can be extended to incorporate the query information. Saggion *et al.* [10] presented a robust summarization system developed within the GATE architecture that makes use of robust components for semantic tagging and co-reference resolution provided by GATE. Wei *et al.* [11] incorporated the query influence into the mutual reinforcement chain to cope with the need for query-oriented multi-document summarization. Wan *et al*. [12] used both relationships among sentences and relationships between the given query and the sentences by manifold ranking. Probability models have also been proposed with different assumptions on the generation process of the documents and the queries [13], [14].

C. Query Expansion

Query expansion is the process of augmenting the user's query with additional terms in order to improve search results. For instance, when we are ready to search "panther" by some search engine, we can expand such query by adding synonyms of "panther" to the query, such as "jaguar," "cougar," etc. Query expansion has also been explored in the field of document summarization, where the quality of the generated summary can be improved. For example, Daume and Marcu [15] propose a justified query expansion technique in the language modeling for IR framework. However, it fails to consider the semantic relatedness between the sentences and the query string.

III. Disaster Management Domain

A. Domain Description

It is well known that hurricanes, earthquakes, and other natural disasters cause immense physical destruction, loss of life and property around the world. The purpose of the disaster management program is to enhance efficient coordination and collaboration among public safety organizations by enabling the interoperable sharing of emergency alerts and incident-related data between disparate systems. One of the disaster management systems aims to analyze the news and reports related to the disaster to provide concise and recapitulative information for domain experts.

B. Domain-Specific Ontology

Generally speaking, an ontology is often provided by domain experts in disaster management domain [1]. Such an ontology provides answers for the questions concerning what entities exist in disaster management, and how such entities can be related within a hierarchy and subdivided according to similarities and differences among them. The ontology described in this paper is related to the domain of hurricane management, involving 109 concepts and 326 concept relations. This ontology is obtained from the disaster management project at Florida International University [26] (http://www.bizrecovery.org). The ontology is created for the purpose of research included in this project, and is provided by the domain experts from the State Emergency Operations Center (EOC)[1] of Florida. The ontology consists of the *Root*, a set of *concepts*, a set of *is-a relations*, a set of *equivalent-class* and a set of *individuals*. A subset of concepts in the ontology hierarchy is shown in Fig. 1.

[1]http://www.floridadisaster.org/eoc/Update/Home.asp.

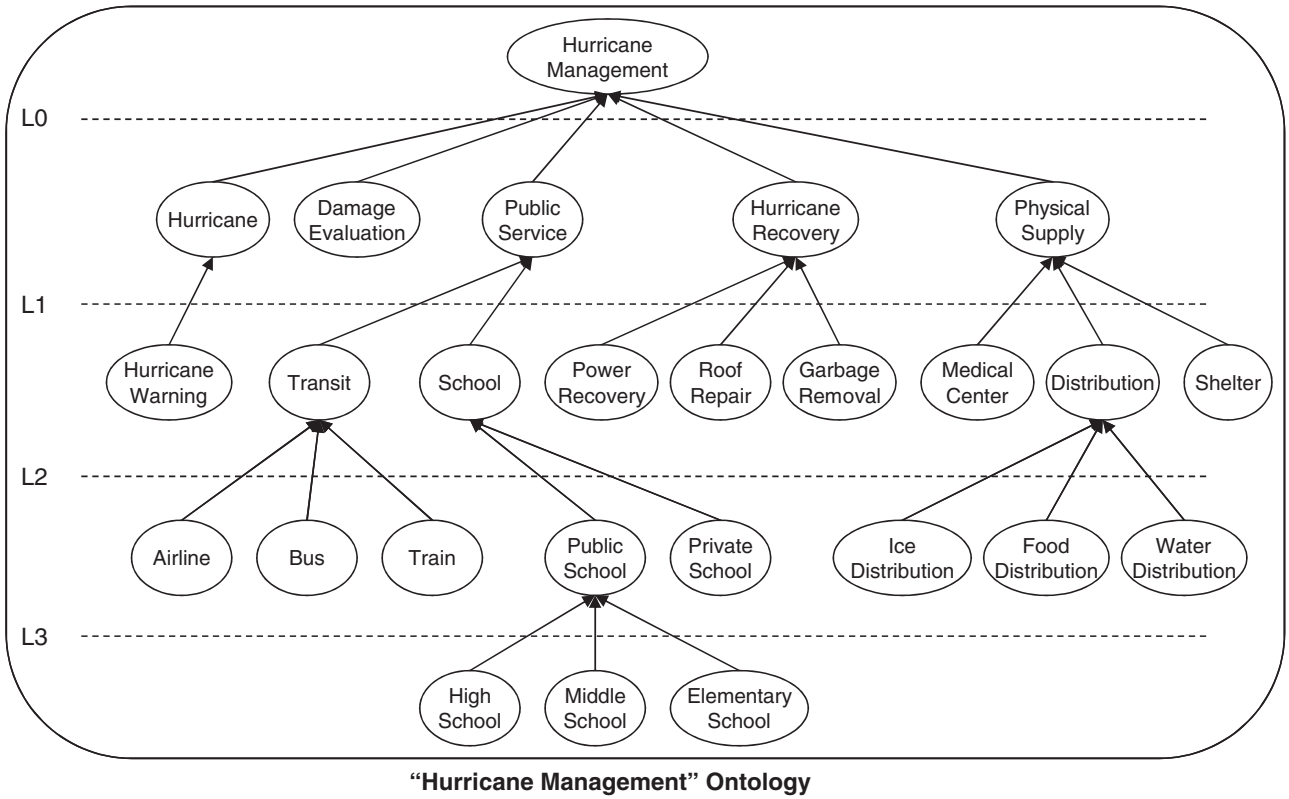


Fig. 1. "Hurricane management" ontology.

IV. Summarization Approaches

To address the summarization issues in the domain of hurricane management, we first map most sentences in the document set onto the domain ontology, and then take advantage of the intrinsic properties of the ontology to represent each sentence. In this section, we explore the effect of the ontology in multi-document summarization tasks from two directions: generic summarization and query-focused summarization.

A. Sentence Mapping

Ontology in disaster management domain provides us abundant conceptual and semantic information, which might facilitate the procedure of multi-document summarization. To utilize the ontology for better understanding the documents, we initially decompose the collection of domain-specific documents into sentences, and then map each sentence to the ontology hierarchy. For each concept of the ontology hierarchy, a group of keywords (i.e., nouns) are assigned by the experts for the sake of sentence mapping. The procedure of sentence mapping is executed based on the following criteria.

1) If the sentence is related to only one concept, map this sentence to the corresponding concept.
2) If the sentence is related to two or more concepts, map this sentence to the least common ancestor (LCA) of these concepts. If the LCA is the most general concept of the ontology, then map the sentence to the original specific concepts.

In this process, we calculate the word set overlapping between a sentence (only considering nouns in the sentence) and the keyword set assigned to each concept as the measure of relatedness, and then rank the scores to select the most related concept. Since different concepts in the ontology have different unambiguous representative noun sets assigned by domain experts, it is unlikely that the same noun will appear in more than one concept. When the condition of the second criterion holds, it means that the sentence contains different words that can map to different concepts. In order to avoid that a single sentence will be linked to multiple concepts and thus make more redundant information, we introduce the LCA of concepts and link the sentence to the LCA if it contains two or more concepts. Based on these criteria, we can guarantee that most sentences are mapped to at least one concept of the ontology because the ontological concepts are representative in a specific domain, and the mapping is reasonable since the mapped sentences can be regarded as instances of the corresponding concepts.

B. Sentence Representation

A key question in multi-document summarization using the ontology is how to represent the sentences we have mapped onto the ontology. We examine several ways to model a sentence into a vector, including term frequency (TF) model [16], term frequency-inverse sentence frequency (TFISF) model [16], term frequency-inverse concept frequency (TFICF) model, concept hierarchy (CH) model [17], [18], and the linear combinations of these models. The vector

space models mentioned above provide different insights for document summarization. In the following, we delve into the details of these models for the purpose of comparison.

Term Frequency Model: In this model, each entry of a sentence vector denotes the term weight (normalized term frequency to prevent a bias toward longer sentences and to give a measure of the importance of the term t_i within the particular sentence s_j). Thus we have the term frequency defined as follows:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{1}$$

where $n_{i,j}$ is the number of occurrences of term t_i in sentence s_j, and the denominator $\sum_k n_{k,j}$ is the sum of number of occurrences of all the terms in sentence s_j.

TFISF Model: Similar to TFIDF [16], term frequency-inverse sentence frequency (TFISF) is used to evaluate how important a word is to a sentence in a corpus. The importance increases proportionally to the number of times a word appears in the sentence but is offset by the frequency of the word in the corpus. The inverse sentence frequency is a measure of the general importance of the term (obtained by dividing the total number of sentences by the number of sentences containing the term, and then taking the logarithm of that quotient), which is defined as

$$isf_i = \log \frac{|S|}{|s : t_i \in s|} \tag{2}$$

where $|S|$ is the total number of sentences in the corpus, and $|s : t_i \in s|$ is the number of sentences where the term t_i appears (that is, $n_{i,j} \neq 0$). Then TFISF is defined as

$$\text{TFISF}_{i,j} = tf_{i,j} * isf_i. \tag{3}$$

A high weight in TFISF is reached by a high term frequency (in the given sentence) and a low sentence frequency of the term in the whole document collection; the weights hence tend to filter out common terms. The TFISF value for a term will always be larger than or equal to zero.

TFICF Model: Term frequency-inverse concept frequency (TFICF) is used to evaluate how important a word is to a concept in an ontology hierarchy. Compared with TFISF, the difference is that the importance increases proportionally to the number of times a word appears under the concept but is offset by the frequency of the concept in the ontology hierarchy. The inverse concept frequency is computed as

$$icf_i = \log \frac{|C|}{|c : t_i \in c|} \tag{4}$$

where $|C|$ is the total number of concepts in the ontology hierarchy, and $|c : t_i \in c|$ is the number of concepts where the term t_i appears (that is, $n_{i,j} \neq 0$). If the term is in the corpus but not appears in any concepts, then the icf value is set to be 0. TFICF is defined as

$$TFICF_{i,j} = tf_{i,j} * icf_i. \tag{5}$$

Concept Hierarchy Model: The is-a taxonomical relations between concepts in the ontology provide apparent hierarchical information, which can be applied to representing a sentence instead of using the whole set of terms/words. In this section, we modify the approach proposed by Yuan and Sun [17], [18]. The new approach, called concept hierarchy (CH) model, is used as an ontology-based structured representation for sentences. The procedure of sentence modeling by CH is as follows.

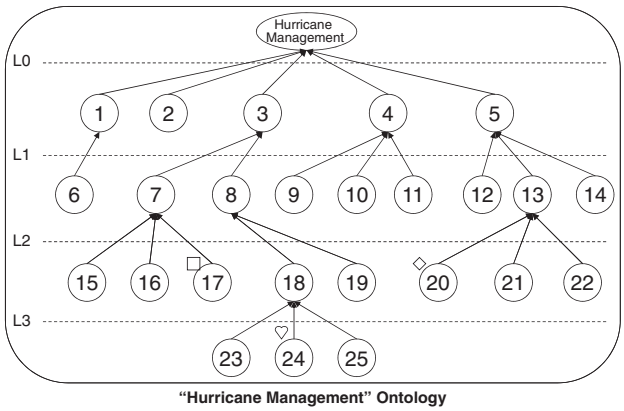


Fig. 2. "Hurricane management" ontology.

1) Given the ontology described in Section III, we number all of the nodes level by level from top to bottom except the root. By doing this, we can obtain an ontology with numbered concepts. Then we map the sentence set onto the ontology based on the criteria described in Section IV-A (Fig. 2).
2) We represent a sentence as a vector V_c with the size of the total number of concepts in the ontology, denoted as $\langle c_1, c_2, \cdots, c_{|C|}\rangle$. This vector is initialized with zeros. The value of each entry c_i is increased by 1 when encountering an existence of the i-th concept node in the sentence. For instance, the term vector of a sentence (related to node 24) unfolds as (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0).
3) To impose the structure into a vector, all of the ancestor nodes of the 1-valued c_i are accounted as existence. Using the previous example of the sentence of node 24, the resulting vector becomes (0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0), as the ancestor nodes of 24 include node 3, 8, 18.
4) V_c is l_2-normalized so that $||V_c||_2 = 1$.

Intuitively, more general concepts in the ontology hierarchy should contribute more to the sentence clustering, since we expect that the sentences can be separated into difference clusters. Consider two sentences that describe two different fine-grained concepts and these two concepts belong to different general concepts, the CH representations of these two sentences are totally different from each other. On the other hand, the fine-grained concepts in the hierarchy would be more influential to differentiate two sentences that are in the same sentence cluster, and therefore might influence the ranking of the sentences.

Linear Combinations: The first three vector space models—TF, TFISF, TFICF—are constructed based on the term feature in the sentences, whereas CH utilizes the concepts appearing

in the whole document collection to represent a sentence. In the following, we try to combine the term-based VSMs and the concept-based VSM together to verify if it is helpful to summarize multiple documents. If we define V_{tf} to be the term frequency vector, V_{tfisf} to be the TFISF vector, V_{tficf} to be the TFICF vector and V_c to be the CH vector, then we can synthesize them as follows:

1) *CH+TF*: $V_{c+tf} = \langle \lambda_1 V_c, (1-\lambda_1) V_{tf} \rangle$;
2) *CH+*TFISF: $V_{c+tfisf} = \langle \lambda_2 V_c, (1-\lambda_2) V_{tfisf} \rangle$;
3) *CH+TFICF*: $V_{c+tficf} = \langle \lambda_3 V_c, (1-\lambda_3) V_{tficf} \rangle$.

Here, λ_1, λ_2, and λ_3 denote the importance of the corresponding concept-based vectors in the combination forms, respectively. The intuition underlying these combinations is that concepts provide an alternative information channel that should be counted separately and weighted independently from any term observations.

C. Generic Summarization

For generic summarization in disaster management domain, the main task in general is to distill the most important overall information from a set of documents related to the disaster. To emphasize the diversity of topic coverage in a generic summary, we employ the standard K-Means method to cluster the sentences of a document collection into different topical groups, and then apply sentence weighting models within each topical group for sentence selection. In addition, we explore the intrinsic properties of the ontology hierarchy to reduce the information redundancy. Fig. 3 represents the generic summarization framework.

Sentence Selection: We apply the centroid-based methods to select important sentences as the summary. To do so, we run the standard k-Means on the sentence set, where the cluster number k is specified as the number of concepts in the first level of the ontology, i.e., the five nodes lying in L_0 in Fig. 1. The intuition is that the diversity of topics in the whole document collection should be restricted in the concept set of the first level of the ontology. Generally speaking, in the domain of disaster management, the analysts often perform post event analysis on different aspects of a single disaster that they might be interested in. By using the ontology, we are able to separate the sentences into different groups, and also the information not mapped directly to the ontology is filtered out by the procedure of sentence mapping, which is not important in terms of post event analysis. To compute the similarity of sentence pairs, we use the cosine similarity [19] based on the vector space models we discussed in Section IV-B. To obtain a reasonable amount of sentences for reading, we select a set of sentences with the total length of L as the candidate sentences from each cluster. Therefore, the cardinality of the candidate sentence set is $k * L$. By this way, we believe that the informative content in the original sentence set can be kept in the candidate sentence set.

Redundancy Reduction: To reduce the information redundancy, we retrieve the concept nodes that correspond to the sentences in the candidate sentence set of each sentence cluster. Then we compare the two concepts of each sentence pair. Here we are concerned with the sentences linked to the same concept in the ontology hierarchy, which is intuitive because the sentences containing the same concept are very likely to describe the same event. Note that the removal of sentences is based on the evolution of the event. Every sentence in the corpus has a timestamp to indicate the time that the event happens or is updated. If the two sentences being considered have close timestamps, for example, the same day, it is very possible that these two sentences are describing the same status of the corresponding event. In such a case, we randomly remove one sentence; otherwise, we keep both of them in the final summary. By doing this, we can maximally reduce the information redundancy of the candidate sentence set of length $k * L$.

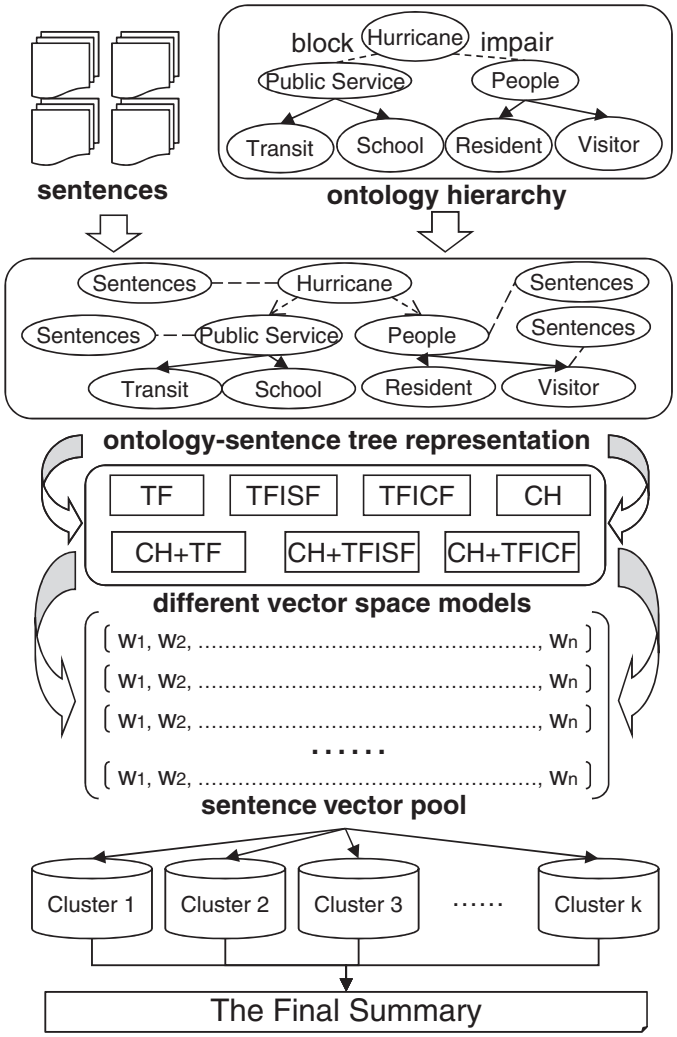


Fig. 3. Generic summarization framework.

Sentence Ranking: After the procedure of reducing information redundancy, we need to rank the remaining sentences to show the importance of each sentence, which is a crucial part of generating text summaries. To this end, we calculate the information content (IC) [20] of the concepts related to each sentence. IC measures the amount of information of a given concept from its probability of occurrence in a corpus. The larger the information content, the more important the concept. IC of a concept is defined as the negation of the

logarithm of the probability $p(a)$ of encountering a concept a in a given corpus as

$$IC(a) = -\log p(a). \tag{6}$$

Resnik [20] proposed the premise that IC of the subsumer must be lower than its specializations, since if the IC associated to each ontological concept does not monotonically increase as concepts are specialized, similarity values would be negatively affected. To guarantee this property, the probability of a concept can be calculated as the sum of the individual occurrences of all the concepts which are subsumed by it as follows:

$$p(a) = \sum_{n \in \text{specializations(a)}} \frac{\text{count(n)}}{N} \tag{7}$$

where specializations(a) is the set of terms subsumed by concept a, and N is the total number of concepts observed in the corpus. To rank the sentences in the candidate sentence set, we compute the IC value of each concept correlated to the sentences in the candidate set, and then sort the sentences under the decreasing order of the IC values. The top ranked sentences are selected as the final summary until the summary length L is reached.

D. Query-Focused Summarization

Query-focused summarization aims at generating a short summary based on a given document set and a given query. The generated summary reflects the condensed information related to the given query within the specified summary length. Given a set of documents and a query, the strategy proposed in Section IV-C can also be applied to query-focused summarization tasks. The only difference is that we need to map the given query to the ontology hierarchy so that we can find the concept related to the query and summarize the sentences attached to this concept and its children. A natural property of the ontology is that many concepts in the ontology have equivalent classes with identical or similar semantic meaning to the concept. This property can serve to expanding the query terms in some cases. Given a set of documents, a domain-specific ontology, and a query, the procedure of query-focused summarization is as follows.

1) First, map all the sentences onto the ontology hierarchy under the same criteria described in Section IV-A. Here we introduce equivalent classes to facilitate the mapping procedure. The equivalent classes of a concept can be treated as implicit relevant nodes of this concept. When mapping a sentence to the ontology, we treat each equivalent class as an explicit node.
2) Second, map the given query onto the ontology. If the query matches multiple concepts on the ontology, we treat them as multiple sub-queries,[2] and combine the generated results for the final summary within a given summary length. If the concepts contained in the query have no equivalent classes, the algorithm will stop expansion.
3) Third, calculate the pairwise concept similarities (Resnik's similarity [20]) between the query classes and the corresponding equivalent class, and then select the most similar equivalent class as the expanded content of the original query.
4) Finally, extract all the sentences linked to the original class and the selected equivalent class as the candidate result set, treat such set as one single cluster, and then compute the centroid of this cluster by using vector space models discussed in Section IV-B. The final query result is formed by the top ranked sentences close to the centroid until the summary length is reached.

[2]If the concepts are in the same subtree of the ontology and have clear hierarchical structure, we treat the result obtained from the most general one as the final summary.

TABLE II
DESCRIPTION OF THE DATASET

Topic	# of documents
Hurricane Information	105
Public Services	268
Social Events	96
Damage Evaluation	47
Hurricane Recovery	438
Physical Supplies	375
Human Management	233
Emergence Management	138

Intuitively, the sentences linked to the equivalent classes are also relevant to the given query, and therefore when summarizing the query-relevant information, these sentences have the opportunity to be selected as the final summary, which can help cover broader semantic information related to the given query. Hence, the final summary can be enriched by the procedure of query expansion to some extent.

V. EXPERIMENTAL RESULTS

A. Real World Data

The document set used in our experiments is a collection of press releases from Miami-Dade County Department of Emergency Management and Homeland Security during Hurricane Wilma from Oct. 19, 2005 to Nov. 4, 2005. It contains 1700 documents in total, concerning all the related events before Wilma came, during Wilma and after Wilma passed. For instance, before Wilma came, a myriad of precaution measures were reported, such as the movement of Wilma, the location of evacuation zones, the canceled social activities, etc. The data used in our experiment differ from general newswire documents in a sense that they contain a lot of routine reporting on multiple aspects of the disaster. Specifically, there are eight major topics in this document set, the description of which is listed in Table II. Note that the topics covered in the documents are summarized by domain experts from EOC, as well as the mapping between the documents and the topics. Each document corresponds to a news article, and when it is released, domain experts assign the article to the relevant topic.

In order to compare the quality of the generated summaries by different approaches, we use human generated summaries as references. For hurricane data, we hire five human labelers

TABLE III
STATISTICS OF THE SENTENCE SET

Total # of sentences	25,016
% of sentences containing one concept	61.4%
% of sentences containing two or more concepts	30.5%

(including one domain expert) to manually create five reference summaries for both generic and query-focused[3] summarization tasks. Under the instruction that the summary should epitomize the major events through the hurricane, the human labelers read all 1700 documents for the reference summaries. The summary length (100 words) is the same for all the compared summaries.

B. Evaluation Metrics

In the domain of disaster management, the domain experts are concerned with the status of different aspects relevant to the disaster. They track the tendency of the events during the disaster and after the disaster passed. Therefore, the expected summary will be several sentences describing the status of some events. In some sense, the format of the summary is similar to the one of general summarization tasks, and therefore we can adopt widely used metrics, for example, ROUGE, to evaluate the quality of the summarization result.

C. Evaluation on Sentence Mapping

Sentence mapping is an important step in our proposed ontology-based method for multi-document summarization. In this section, we provide some empirical evaluation on the efficacy of this procedure. The data used in this experiment includes all 1700 documents of our hurricane dataset. Table III shows some statistics after we decompose documents into sentences. Note that besides 91.9% of sentences containing concepts defined in the ontology, 8.1% of sentences do not contain any concept, and therefore this type of sentences would be filtered out by the procedure of sentence mapping.

To evaluate the efficacy of sentence mapping procedure, we record the percentages of different types of sentences after mapping, i.e., sentences mapped to the direct concept, sentences mapped to the LCA etc. We then randomly choose a subset of sentences (100 sentences) from each type of sentence set, and manually evaluate the quality of the mapped sentences. Table IV describes the statistics, associated with the quality measurement (accuracy[4]) by human evaluation.

Most sentences in the dataset are mapped to at least one concept in the ontology, and the quality of the mapping is acceptable under manual evaluation.

D. Generic Summarization

For the generic summarization, we evaluate different vector space representations for comparison. Note that since we have eight general categories at the top level of the domain ontology, we set the length of candidate sentence set as $8 * 100$, which means in each category, a summary with length of 100 words is generated. After removing redundancy and ranking sentences, we iteratively select the top ranked sentences as our final summary, until the length reaches to 100. In order to get better performance, we need to figure out the importance factor λ_1, λ_2, and λ_3, which may result in different experimental results as discussed in Section IV-B. To do so, we conduct experiments on a subset of documents[5] to evaluate the sensitivity of the parameters in different combinations, using ROUGE-2 value as the metric. The factor evaluation results are shown in Fig. 4.

TABLE IV
EVALUATION OF SENTENCE MAPPING

Sentence type	Percentage	Accuracy
Sentences mapped to one direct concept	60.8%	96.5%
Sentences mapped to two or more concepts	10.3%	97.9%
Sentences mapped to the LCA	20.0%	94.2%
Sentences filtered out	8.9%	98.1%
Sentences mapped to at least one concept per document on average	90.4%	–

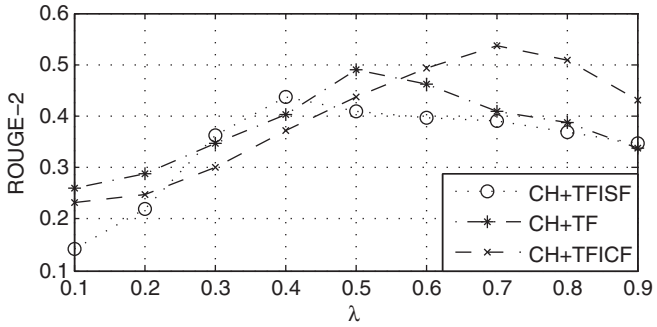


Fig. 4. ROUGE-2 for evaluating λ.

We set λ_1 as 0.5 for *CH+TF*, λ_2 as 0.4 for *CH*+TFISF, and λ_3 as 0.7 for *CH+TFICF*. We implement the following widely used or recent published methods for generic summarization to compare with the methods we discussed in Section IV-C.

1) MEAD [7]: extracts sentences based on centroid value, positional value and first sentence overlap.
2) Latent Semantic Analysis (LSA) [21]: identifies semantically important sentences by conducting latent semantic analysis.
3) Nonnegative Matrix Factorization (NMF) [4]: performs NMF on sentence-term matrix and selects the high ranked sentences.
4) LexPageRank [5]: first constructs a sentence connectivity graph based on cosine similarity and then selects important sentences based on the concept of eigenvector centrality [5].
5) BSTM [22]: a Bayesian sentence-based topic model making use of both the term-document and term-sentence associations.

As for the methods which generate nondeterministic results, we run them 10 times and calculate the average ROUGE score.

[3]The query information is described in Section V-E.

[4]The accuracy here means that the percentage of sentences that belong to the corresponding sentence type over the sentence set.

[5]It contains 400 documents related to hurricane management. We use the remaining documents for summarization.

TABLE V
RESULTS ON GENERIC SUMMARIZATION

System	ROUGE-SU4	ROUGE-L	ROUGE-S*	ROUGE-2
MEAD	0.35152	0.43113	0.12821	0.13541
LSA	0.23611	0.38356	0.09689	0.10461
NMF	0.36478	0.44720	0.13812	0.14576
LexPageRank	0.22642	0.37267	0.11500	0.12167
BSTM	0.45113	0.59259	0.21916	0.22897
TFISF	0.13115	0.28109	0.06490	0.06974
TF	0.22667	0.36842	0.08862	0.09580
TFICF	0.23030	0.37126	0.12299	0.12946
CH	0.32381	0.47170	0.19948	0.20460
CH+TFISF	*0.43166*	*0.56738*	*0.19805*	*0.20724*
CH+TF	*0.47682*	*0.58438*	*0.26355*	*0.27256*
CH+TFICF	**0.54546**	**0.60162**	**0.30827**	**0.31607**

TABLE VI
RESULTS ON QUERY-FOCUSED SUMMARIZATION

System	ROUGE-SU4	ROUGE-L	ROUGE-S*	ROUGE-2
Qs-MRF	0.35676	0.45913	0.16861	0.18059
Wiki	0.33290	0.44835	0.15378	0.17542
SNMF	0.47279	0.51314	0.19806	0.21032
CH+TFISF	0.42034	0.55607	0.18924	0.19442
CH+TF	0.46743	0.57238	0.25205	0.26857
CH+TFICF	0.53290	0.59061	0.29437	0.30549
CH+TFISF(Q)	*0.47825*	*0.51805*	*0.19746*	*0.21653*
CH+TF(Q)	*0.52360*	*0.55019*	*0.25896*	*0.27258*
CH+TFICF(Q)	**0.60058**	**0.60732**	**0.31640**	**0.32908**

Table V presents the experimental comparison of different generic summarization approaches. Bold means the result is statistically significant at the 99% level.

From the comparison results, we observe that: (1) The methods which take into consideration the concepts appearing in the document collection are better than the traditional approaches; (2) The methods using the combinations of different VSMs perform better than the ones solely using one VSM, since the information relevant to the concept hierarchy is integrated into the VSMs, which enriches the representation pattern from both macroscopic and microscopic perspectives; (3) The method with *CH+TFICF* vector representation clearly outperforms the other rivals. Upon the above analysis, we conclude that for the generic summarization in disaster management, the concepts contained in the disaster-related document collection provide substantial benefits for our summarization results.

In order to evaluate how other system designs, such as sentence mapping and redundancy removal, can impact the summarizer, we conduct the experiments as follows:

1) for sentence mapping, we compare the quality of the summaries using LCA and mapping the sentence to different concepts if it contains multiple concepts; and
2) for redundancy removal (RR), we try to evaluate the results with redundancy removal and without redundancy removal. We run the experiments ten times and average the ROUGE-2 scores. Table VII shows the comparison result.

From the comparison, we can observe the following:

1) sentence mapping using Least Common Ancestor can help improve the quality of the summary to some extent, compared with the design that maps the sentences to the corresponding concepts instead of LCA; and
2) redundancy removal can indeed provide high-quality summaries.

TABLE VII
EVALUATION ON DIFFERENT SYSTEM DESIGNS IN GENERIC SUMMARIZATION

System	No LSA	LSA	No RR	RR
CH	0.28175	0.32294	0.22907	0.32350
CH+TFISF	0.41291	0.43215	0.34729	0.43189
CH+TF	0.44706	0.47593	0.40068	0.47626
CH+TFICF	0.51137	0.54431	0.46715	0.54609

E. Query-Focused Summarization

For query-focused summarization, a list of query strings (20 queries) are manually generated for the purpose of the query action. In general, the specified queries contain at least one of the concepts in the ontology hierarchy, so that the proposed method can automatically locate the position of the concept on the ontology. To get the best summarization results, we continue to use the important factors obtained in the generic summarization. We compare the method we discussed in Section IV-D with some widely used and recently published systems.

1) Qs-MRF [11]: extends the mutual reinforcement principle between sentence and term to document-sentence-term mutual reinforcement chain, and uses query-sensitive similarity to measure the affinity between the pair of texts.
2) Wiki [8]: uses Wikipedia as external knowledge to expand query and builds the connection between the query and the sentences in documents.
3) SNMF [4]: calculates sentence-sentence similarities by sentence level semantic analysis, clusters the sentences via symmetric non-negative matrix factorization, and extracts the sentences based on the clustering result. Table VI presents the experimental comparison of different query-focused summarization approaches, Bold means the result is statistically significant at the 99% level. Table VIII lists a set of sample queries and the corresponding system summaries.

Note that "CH+TFISF(Q)," "CH+TF(Q)," and "CH+TFICF(Q)" represent the methods with different VSMs combined with the procedure of query expansion. From the comparison, we have two observations:

1) the methods with query expansion outperform other recently published systems;
2) query expansion enriched summarization methods do improve the summary quality in disaster management domain, particularly, by using the sentence representation of "CH + TFICF." It is straightforward that the procedure of query expansion serves to supplying great opportunity for more sentences relevant to the given query to be selected into the final summary. Therefore, the summarization results are enriched by query expansion in terms of summary quality.

TABLE VIII
SAMPLE QUERIES AND SUMMARIES

Query	Description	CH+TFICF-based Summary
1	What is the *transit* status?	**Metrobus** has experienced increased demand. **Buses** will operate along the metrorail alignment free of charge. **Parking** at **metrorail stations** will also be free...
2	Is the *airport* open these days?	Homestead **airport** remains closed. passengers should continue checking with their **airlines** regarding specific **flights**...
3	*Evacuation center* status.	Some **distribution points** have been closed. Residents should not gather at designated **distribution centers** unless they have checked for availability...
4	*Schools* at Miami-Dade county.	**Public schools** are closed Monday and Tuesday. Miami dade **public schools** will be open tomorrow : Thursday, October 20...

VI. CONCLUSION AND FUTURE WORK

In this paper, we gave an empirical study on several approaches that utilize the ontology to solve different multidocument summarization problems in disaster management domain. For generic summarization, we employed different vector space models to represent sentences in the document collection, and explored the feasibility of different combinations of the VSMs. Then the centroid-based methods were utilized to cluster the sentence set and the important sentences close to the centroids of the sentence clusters are extracted. The final summary was subsequently generated by reducing information redundancy and ranking sentences. For query-focused summarization, we delved into the effect of query expansion in summarization tasks. The ontology is rich in conceptual information related to the specific domain. We will keep working on the issue of ontology-based multidocument summarization, particulary, on some other document summarization tasks, i.e., update summarization and comparative summarization. Another interesting direction is to explore deeply how to utilize the hierarchical correlations in the ontology to further improve the quality of the summary and to perform hierarchical text categorization [24], [23]. In addition, we will try to employ information extraction techniques to further improve summarization results. We are also interested in extending our proposed method to the summarization using public ontologies, for example, WordNet and Wikipedia. The generality and scalability issues should be taken into account for further extension.

REFERENCES

[1] E. Klien, M. Lutz, and W. Kuhn, "Ontology-based discovery of geographic information services—An application in disaster management," *Comput., Environ. Urban Syst.*, vol. 30, no. 1, pp. 102–123, 2006.

[2] H. Hsu, C. Tsai, M. Chiang, and C. Yang, "Topic generation for web document summarization," in *Proc. IEEE SMC*, 2008, pp. 3702–3707.

[3] X. Yong-dong, W. Xiao-long, L. Tao, and X. Zhi-ming, "Multi-document summarization based on rhetorical structure: Sentence extraction and evaluation," in *Proc. IEEE SMC*, 2008, pp. 3034–3039.

[4] D. Wang, T. Li, S. Zhu, and C. Ding, "Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization," in *Proc. SIGIR*, 2008, pp. 307–314.

[5] G. Erkan and D. Radev, "Lexpagerank: Prestige in multi-document text summarization," in *Proc. EMNLP*, vol. 4, 2004, pp. 365–371.

[6] X. Wan and J. Yang, "Multi-document summarization using cluster-based link analysis," in *Proc. SIGIR*, 2008, pp. 299–306.

[7] D. Radev, H. Jing, M. Sty, and D. Tam, "Centroid-based summarization of multiple documents," *Inf. Process. Manage.*, vol. 40, no. 6, pp. 919–938, 2004.

[8] V. Nastase, "Topic-driven multi-document summarization with encyclopedic knowledge and spreading activation," in *Proc. EMNLP*, 2008, pp. 763–772.

[9] C. Lee, Z. Jian, and L. Huang, "A fuzzy ontology and its application to news summarization," *IEEE Trans. Syst., Man, Cybern., B Cybern.*, vol. 35, no. 5, pp. 859–880, Oct. 2005.

[10] H. Saggion, K. Bontcheva, and H. Cunningham, "Robust generic and query-based summarisation," in *Proc. ECAL*, 2003, pp. 235–238.

[11] F. Wei, W. Li, Q. Lu, and Y. He, "Query-sensitive mutual reinforcement chain and its application in query-oriented multi-document summarization," in *Proc. SIGIR*, 2008, pp. 283–290.

[12] X. Wan, J. Yang, and J. Xiao, "Manifold-ranking based topic-focused multi-document summarization," in *Proc. IJCAI*, 2007, pp. 2903–2908.

[13] J. Tang, L. Yao, and D. Chen, "Multi-topic based query-oriented summarization," in *Proc. SDM*, 2009.

[14] A. Haghighi and L. Vanderwende, "Exploring content models for multi-document summarization," in *Proc. HLT-NAACL*, 2009, pp. 362–370.

[15] H. Daumé and D. Marcu, "Bayesian query-focused summarization," in *Proc. ACL*, vol. 44, no. 1. 2006, p. 305.

[16] D. Jurafsky, J. Martin, A. Kehler, K. Vander Linden, and N. Ward, *Speech And Language Processing*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2000.

[17] S.-T. Yuan and J. Sun, "Ontology-based structured cosine similarity in speech document summarization," in *Proc. WI*, 2004, pp. 508–513.

[18] S. Yuan and J. Sun, "Ontology-based structured cosine similarity in document summarization: With applications to mobile audio-based knowledge management," *IEEE Trans. Syst., Man, Cybern., B Cybern.*, vol. 35, no. 5, pp. 1028–1040, Oct. 2005.

[19] P. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*. Boston, MA, USA: Pearson Addison Wesley, 2006.

[20] P. Resnik, "Using information content to evaluate semantic similarity in a taxonomy," Arxiv preprint cmp-lg/9511007, 1995.

[21] Y. Gong and X. Liu, "Generic text summarization using relevance measure and latent semantic analysis," in *Proc. SIGIR*, 2001, pp. 19–25.

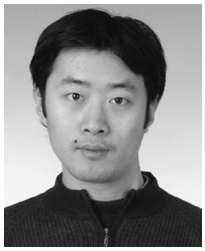
[22] D. Wang, S. Zhu, T. Li, and Y. Gong, "Multi-document summarization using sentence-based topic models," in *Proc. ACL-IJCNLP*, 2009, pp. 297–300.

[23] T. Li, S. Zhu, and M. Ogihara. "Text categorization via generalized discriminant analysis," *Inf. Process. Manage.*, vol. 44, no. 5, pp. 1684–1697, 2008.

[24] T. Li, S. Zhu, and M. Ogihara. "Hierarchical document classification using automatically generated hierarchy," *J. Intell. Inf. Syst.*, vol. 29, no. 2, pp. 211–230, 2007.

[25] L. Li, D. Wang, C. Shen, and T. Li, "Ontology-enriched multi-document summarization in disaster management," in *Proc. SIGIR*, 2011, pp. 819–820.

[26] L. Zheng, C. Shen, L. Tang, T. Li, S. Luis, S. Chen, and V. Hristidis, "Using data mining techniques to address critical information exchange needs in disaster affected public-private networks," in *Proc. SIGKDD*, 2010, pp. 125–134.



Lei Li received the M.S. degree in software engineering from Beihang University, Beijing, China, in 2008. He is currently pursuing the Ph.D. degree with Florida International University, Miami, FL, USA.

His current research interests include data mining, machine learning, information retrieval, and building recommender systems.

Tao Li received the Ph.D. degree in computer science from the Department of Computer Science, University of Rochester, Rochester, NY, USA, in 2004.

He is currently an Associate Professor with the School of Computing and Information Sciences, Florida International University, Miami, USA. His current research interests include data mining, computing system management, information retrieval, and machine learning.

Dr. Li was a recipient of the NSF CAREER Award and multiple IBM Faculty Research Awards.

Data Mining Meets the Needs of Disaster Information Management

Li Zheng, Chao Shen, Liang Tang, Chunqiu Zeng, Tao Li, Steve Luis, and Shu-Ching Chen, *Senior Member, IEEE*

***Abstract*—Techniques to efficiently discover, collect, organize, search, and disseminate real-time disaster information have become national priorities for efficient crisis management and disaster recovery tasks. We have developed techniques to facilitate information sharing and collaboration between both private and public sector participants for major disaster recovery planning and management. We have designed and implemented two parallel systems: a web-based prototype of a Business Continuity Information Network system and an All-Hazard Disaster Situation Browser system that run on mobile devices. Data mining and information retrieval techniques help impacted communities better understand the current disaster situation and how the community is recovering. Specifically, information extraction integrates the input data from different sources; report summarization techniques generate brief reviews from a large collection of reports at different granularities; probabilistic models support dynamically generating query forms and information dashboard based on user feedback; and community generation and user recommendation techniques are adapted to help users identify potential contacts for report sharing and community organization. User studies with more than 200 participants from EOC personnel and companies demonstrate that our systems are very useful to gain insights about the disaster situation and for making decisions.**

***Index Terms*—Data mining, disaster information management, dynamic query form, hierarchical summarization, user recommendation.**

I. Introduction

BUSINESS closures caused by disasters can cause millions of dollars in lost productivity and revenue. A study in Contingency Planning and Management shows that 40% of companies that were shut down by a disaster for three days failed within 36 months. Thin margins and a lack of a well-designed and regularly tested disaster plan can make companies, particularly small businesses, especially vulnerable [1]. We believe that the solution to better disaster planning and recovery is one where the public and private sectors work together to apply computing tools to deliver the right information to the right people at the right time to facilitate the work of those working to restore a community's sense of normalcy. While improved predictive atmospheric and hydrological models and higher quality of building materials and building codes are being developed, more research is also necessary for how to collect, manage, find, and present disaster information in the context of disaster management phases: preparation, response, recovery, and mitigation [4], [30].

In the United States, the Federal Emergency Management Agency (FEMA) has recognized the importance of the private sector as a partner in addressing regional disasters. The State of Florida Division of Emergency Management has created a Business and Industry Emergency Support Function designed to facilitate logistical and relief missions in affected areas. Four counties, Palm Beach, Broward, Miami-Dade, and Monroe, which constitute the Southeastern population of South Florida and include over 200 000 business interests, are developing Business Recovery Programs to help facilitate faster business community recovery through information sharing and collaboration.

Disaster management researchers at Florida International University have collaborated with the Miami-Dade Emergency Operations Center (EOC), South Florida Emergency Management and industry partners including Wal-Mart, Office Depot, Wachovia, T-Mobile, Ryder Systems, and IBM to understand how South Florida public and private sector entities manage and exchange information in a disaster situation. The efficiency of sharing and management of information plays an important role in the business recovery in a disaster [3]. Users are eager to find valuable information to help them understand the current disaster situation and recovery status. The community participants (the disaster management officials, industry representatives, and utility agents) are trying to collaborate to exchange critical information, evaluate the damage, and make a sound recovery plan. For example, it is critical that companies receive information about their facilities, supply chain, and city infrastructure. They seek this information from media outlets like television/radio newscasts, employee reports, and conversations with other companies with which they have a relationship. With so many sources of information, with different levels of redundancy and accuracy, possibly generated by a variety of reports (structured and unstructured), it is difficult for companies to quickly assimilate such data and understand their situation.

We have learned that a large-scale regional disaster may cause a disruption in the normal information flow, which in turn affects the relationships between information producers and consumers. Effective communication is critical in a crisis situation.

Manuscript received February 28, 2013; revised April 4, 2013, June 1, 2013, and July 10, 2013; accepted August 12, 2013. Date of publication October 4, 2013; date of current version October 16, 2013. The work was supported in part by the National Science Foundation under grants HRD-0833093, CNS-1126619, and IIS-1213026, the U.S. Department of Homeland Security under grant Award Number 2010-ST-062–000039, and Army Research Office under grant number W911NF-10–1–0366 and W911NF-12-1-0431. This paper was recommended by Associate Editor S. Rubin.

The authors are with the School of Computing and Information Sciences, Florida International University, Miami, FL 33199 USA (e-mail: lzhen001@cs.fiu.edu; cshen001@cs.fiu.edu; ltang002@cs.fiu.edu; czeng001@cs.fiu.edu; taoli@cs.fiu.edu; luiss@cs.fiu.edu; chens@cs.fiu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/THMS.2013.2281762

What is not very well known is how to effectively discover, collect, organize, search, and disseminate real-time disaster information.

Our study of the hurricane disaster information management domain has revealed two interesting yet crucial information management issues that may present similar challenges in other disaster management domains. The first issue is that reconstructing or creating information flow becomes intractable in domains where the stability of information networks is fragile and can change frequently. However, important information networks often carry and store critical information between parties, which dominates the flow of resources and information exchanges. The consequence is that the ability and the efficiency of communication degrade once critical networks are disrupted by the disaster and people may not have alternative paths to transfer information. For example, once power is disabled and uninterruptable power supplies fail after a hurricane, computing and networking equipment will fail unless preventative measures are taken. However, maintaining a fuel-consuming generator is not always possible.

Another issue is the large volume of disaster situation information. Reading and assimilating situational information are very time consuming and may involve redundant information. Thus, to quickly reassemble or create information flows for multiparty coordination activities during disaster situations, technologies that are capable of extracting information from recent updates, delivering that information without conflict or irrelevance, and representing preferential information are needed.

This research is mainly focused on the second issue. Research in disaster management addresses the needs and challenges of information management and decision making in disaster situations [36]–[38]. We have developed an understanding of those needs for hurricane scenarios. The information delivery should support users' complex information needs tailored to the situation and the tasks; and the information should be synthesized from heterogeneous sources and tailored to specific contexts or tasks at hand. It should be summarized for effective delivery and immediate usefulness for making decision.

A. Related Work

The approaches and the tools that are used for information sharing vary based on the task and scale of the participating agencies or the types of information exploration platforms.

Commercial systems, such as WebEOC [39] and E-Teams [40] used by Emergency Management departments located in urban areas, can access multiple resources. A Disaster Management Information System developed by the Department of Homeland Security is available to county emergency management offices and participating agencies to provide an effective reports/document sharing software system. The National Emergency Management Network [41] allows local government to share resources and information about disaster needs; The RESCUE Disaster Portal is a web portal for emergency management and disseminating disaster information to the public [4]; The Puerto Rico Disaster Decision Support Tool is an Internet-based tool for disaster planners, responders, and related officials at the municipal, zone, and state level for access to a variety of geo-referenced information [35].

Efforts, such as GeoVISTA [31], facilitate the information distribution process in disasters. GeoVISTA monitors tweets to form situation alerts on a map-based user interface according to the geo-location associated with the tweets. Such a system applies geographic information sciences to scientific, social, and environmental problems by analyzing geospatial data [31].

These useful situation-specific tools provide query interfaces, and GIS and visualization capabilities to simplify the users' interaction and convey relevant information. The primary goal of these systems are message routing, resource tracking, and document management for the purpose to support situation awareness, demonstrate limited capabilities for automated aggregation, data analysis, and mining [4].

However, these tools do not consider how different communities interact with other businesses and county organizations. Further, these tools do not allow for the integration of real-time information. They do not provide information extraction (IE), information retrieval (IR), information filtering (IF), and data mining (DM) techniques needed when delivering personalized situation information to different types of users.

B. Design Challenges

We have identified four key design challenges for disaster information sharing platforms and tools.

1) *Effective techniques to capture the status information:* Participants need to communicate status through many channels, including email, mailing lists, web pages, press releases, and conference calls. It is desirable to capture such status information when it is available and to prevent redundant reporting. To facilitate the reuse of such materials, users should be able to update status information via unstructured documents such as plain text, Adobe PDFs, and documents. It is necessary to identify the useful information in the documents.

2) *Effective and interactive information summarization methods:* It is important to build a summarized view to support understanding the situation from reports. Multidocument summarization provides users with a tool to effectively extract important and related ideas of current situations. Previous text summarization techniques gave users a fixed set of sentences based on the user query. An interactive summarization interface is needed to help users navigate collected information at different granularities, and locate their target information more efficiently.

3) *Intelligent information delivery techniques:* Data can be collected through different channels and may belong to different categories. During disaster preparation and recovery, users do not have the time to go through the system to find the information they want. Structured information can help people make decisions by providing them with actionable and concrete information representation and exploration. However, navigating large datasets on a mobile device is particularly inefficient. An interactive tabular interface can help users filter useful information by adaptively changing query conditions and user feedback.

4) *Dynamic community generation techniques:* In information sharing tasks, identifying a group of recipients to which a certain type of information is conveyed can improve the efficiency of communication. In addition, identifying how participants interact with these communities in a disaster situation may reveal information helpful in a recovery scenario. User recommendation techniques can automatically and interactively generate potential recipients for different pieces of information. In addition, user recommendation techniques can help to dynamically organize user groups according to various information sharing tasks.

We created an information-rich service on both web-based and mobile platforms in the disaster management domain to address the design challenges. In particular, to address the first challenge, we apply information extraction to automatically extract the status information from documents. To address the second challenge, we apply hierarchical summarization to automatically extract the status information from a large document set and also provide a hierarchical view to help users browse information at different granularities. To address the third, we create a user interface capability called the dynamic dashboard to improve information quality to match user's interests, and use document summarization techniques to give users fast access to multiple reports. In addition, a dynamic query form is designed to improve information exploration quality on mobile platforms. It captures users' interests by interactively allowing them to refine and update their queries. To address the fourth challenge, for community discovery, we adopt spatial clustering techniques to track assets like facilities, or equipment, which are important to participants. The geo-location of such participants can be organized into dynamic communities, and these communities can be informed about events or activities relevant to their spatial footprints. For user recommendation, we use transactional recommendation history combined with textual content to explore the implicit relationship among users.

Thus, we designed and implemented a web-based prototype of a Business Continuity Information Network (BCiN) that is able to link participating companies into a community network, provide businesses with effective and timely disaster recovery information, and facilitate collaboration and information exchange with other businesses and government agencies. We also designed and implemented an All-Hazard Disaster Situation Browser (ADSB) system that runs on Apple's mobile operating system (iOS), and iPhone and iPad mobile devices. Both systems utilize the data processing power of advanced information technologies for disaster planning and recovery under hurricane scenarios. They can help people discover, collect, organize, search, and disseminate real-time disaster information [4], [5].

This study introduces a unified framework that systematically integrates the different techniques developed in [5] and [29]. The idea is that such a framework can be utilized when dealing with different systems or applications separately (e.g., BCiN and ADSB), and hopefully can be easily applied to other scenarios having critical information sharing and management needs.

The rest of the paper is organized as follows. Section II describes the system architecture: information extraction techniques to create structured records, the hierarchical summarization module, the dynamic dashboard and the dynamic query form modules, and community identification and user recommendation modules. Section III presents two case studies of the BCiN and ADSB systems. Section IV describes the system evaluation and data crawling strategies. The conclusion is in Section V.

TABLE I
EXAMPLE OF EOC REPORT

Time: October 21, 2005 12:30 p.m.
Miami-Dade Emergency Operations Center is currently activated at a level II and officials and emergency managers are carefully monitoring Hurricane Wilma.
Residents are urged to finalize their personal hurricane preparations.
On Monday, October 24, Miami-Dade County offices, public schools, and courts will be closed.
Currently, transit bus and rail service continues, including Metrobus, Metrorail and Metromover.
Miami International Airport is open. However, if you have travel plans please check with your airline for flight information.
Tomorrow afternoon, the American Red Cross will open hurricane evacuation centers for residents who do not feel safe in their homes or live in low-lying areas.

II. SYSTEM ARCHITECTURE

A. Structured Information Extraction From Reports

A user interface supports information sharing among companies and government agencies. We do not request a unified format for them to submit the reports. Instead, we use information extraction methods to integrate reports from different sources. For example, Table I shows an example of EOC reports.

The key information is "What was/is/will be the status of Facilities/Services/... at the time of ...." From the EOC reports, we need to extract such information in the form of a triple, entity, time, and status, which reveals the status information of the entity at a certain time. In EOC reports, the entity may be a facility or public service like "Miami International Airport," "schools," "bus," and an order like "curfew." If the entity represents an order, the triple means whether the order is in effect (or not) at that specific time. We extract these triples through two steps: first, we extract entities and time expressions, then, we classify a pair of (service, time) to a proper category, "no relation"/"open"/"close"/"unclear." We assume that the information of one event is described in one sentence, so we process every sentence individually to extract an event. To extract those triples, both entity and relation extraction will be performed. Sometimes two different reports generate the same events, which have the same extracted information, such as the same hurricane name, the same date and the same status of traffic. The repeated events will be deleted. Note that the date/time is an important attribute for every event. Two events with different date/time (at the hourly level) are treated as two different events.

1) Entity Extraction: For each report, sentence segmentation is conducted, and each sentence is Part-Of-Speech-tagged [34]. To extract entities and time expressions, we manually label some news and train a linear chain conditional random fields model to

TABLE II
ENTITY EXTRACTION RESULT OF THE REPORT IN TABLE I

Miami-Dade Emergency Operations Center is currently activated at a level II and officials and emergency managers are carefully monitoring Hurricane Wilma.
Residents are urged to finalize their personal hurricane preparations.
On <T>Monday, October 24</T>, <E>Miami-Dade County offices</E>, <E>public schools</E>, and <E>courts</E> will be closed.
<T>Currently</T>, <E>transit bus</E> and <E>rail service</E> continues, including <E>Metrobus</E>, <E>Metrorail</E> and <E>Metromover</E>.
<E>Miami International Airport</E> is open. However, if you have travel plans please check with your airline for flight information.
<T>Tomorrow afternoon</T>, the American Red Cross will open <E>hurricane evacuation centers</E> for residents who do not feel safe in their homes or live in low-lying areas.

tag all words, using "BIO" annotation [6], [7]. A word tagged as [TYPE-B]/[TYPE-I] means it is the beginning/continuing word of the phrase of the TYPE, and the word tagged by O means it is not in any phrase. TYPE can be E for entity or T for time expression. Given sentence X, the probability that its tags are Y is

$$p(Y|X) = \frac{1}{Z_X} \exp\left(\sum_{i,k} \lambda_k f_k(y_{i-1}, y_i, X) + \sum_{i,l} \mu_l g_l(y_{i-1}, y_i, X)\right) \quad (1)$$

where Z_X is the normalization constant that makes the probability of all state sequences sum to one; $f_k(y_{i-1}, y_i, X)$ is an arbitrary feature function over the entire observation sequence and the states at positions i and $i-1$, while $g_l(y_{i-1}, y_i, X)$ is a feature function of the states at position i and the observation sequence; and λ_k and μ_l are the weights learned for the feature functions f_k and g_l, reflecting the confidence of feature functions by maximum likelihood procedure. The most probable labels can be obtained as

$$Y^* = \underset{Y}{\operatorname{argmax}}\, P(Y|X) \quad (2)$$

by the Viterbi-like dynamic programming algorithm [6]. Features that we use are the local lexicons and POS tags, and the dictionary composed of the existent entity names in the database. Table II shows the entity extraction results of the report in Table I.

2) Relation Extraction: If a sentence contains an entity but no time expression, the time of the report will be associated with the sentence. To generate the triple by connecting the entity with the time expression with a proper status label, we train a multicategory support vector machine [8] to classify each pair of (entity, time) to a proper category, defined as "no relation"/"open"/"close"/"unclear." Table III shows the features that we used for classification, from which the TenseOfSentence(*e,t*), NegativeVerbsInSentence(*e,t*), and PositiveVerbsInsentence(*e,t*) are extracted as the heuristic rules to indicate the tense of the sentence, the verbs with negative modifiers, and the verbs without negative modifiers semantically in the sentence. Note that FromDocument(t) indicates whether the time is the time associated with document.

TABLE III
FEATURES USED TO CLASSIFY WHETHER THE ENTITY e IS ASSOCIATED WITH THE TIME EXPRESSION t

DistanceBetween (e, t)
WordBetween(e,t)
TenseOf Sentence(e,t)
NegativeVerbsInSentence(e,t)
PositiveVerbsInsentence(e,t)
ContainDate(t)
PrepositionBefore(t)
FromDocument(t)

TABLE IV
INFORMATION EXTRACTED FROM THE EOC REPORT IN TABLE I

Service	Time	Status
Miami-Dade County offices	October 24, 2005	close
public schools	October 24, 2005	close
courts	October 24, 2005	close
transit bus	October 22, 2005 6:30 p.m.	open
Rail service	October 22, 2005 6:30 p.m.	open
…		
Miami International Airport	October 22, 2005 6:30 p.m.	open
hurricane evacuation centers	October 23, 2005 afternoon	open

We extract those pairs of entity and time expressions in "open" and "close" categories to form the triple. The time expressions are formatted into an absolute form of expression from relative time expressions such as "next Monday," "this afternoon" using the time of the report as a benchmark. The structured information that is extracted from the report in Table I is shown in Table IV.

B. Report Summarization

The hierarchical multidocument summarization method generates the hierarchical summaries of reports. We use the affinity propagation (AP) [9] clustering method to build a hierarchical structure for sentences of related reports.

1) Affinity Propagation: The input of the AP algorithm is the sentence similarity graph defined as *G<V, E>*: V is the set of vertices with each vertex, called data point, representing a sentence. E is the set of edges. Let $s(i,k)$ be the similarity between two distinct points i and k, indicating how well data point k is suitable to be the exemplar of point i. Especially, $s(i,i)$ is the preference of a sentence i to be chosen as the exemplar. There are two kinds of messages passing between data points: responsibility and availability.

The responsibility $r(i,k)$ is computed as follows:

$$r(i,k) \leftarrow s(i,k) - \max_{\{k' \neq k\}} \{a(i,k') + s(i,k')\}. \quad (3)$$

The responsibility $r(i,k)$ is passing from i to candidate exemplar k. It reflects the accumulated evidence of how well point k is selected as the exemplar for point i against other candidate exemplars.

The availability $a(i,k)$ is computed as follows:

$$a(i,k) \leftarrow \min\{0, r(k,k) + \sum_{i' \in V, i' \notin \{i,k\}} \max\{0, r(i',k)\}\}. \tag{4}$$

The availability $a(i,k)$ is passing from the candidate exemplar k to point i, reflecting the accumulated evidence of how appropriate point i to choose point k as its exemplar, considering the support from other points that share point k as exemplar, whereas the responsibility updating lets all candidate exemplars compete for the ownership of a data point, the availability updating gathers evidence from data points to measure the goodness of each candidate exemplar.

The self-availability $a(k,k)$ is updated as follows:

$$a(k,k) \leftarrow \sum_{i' \in V, i' \neq k} \max\{0, r(i',k)\}. \tag{5}$$

This message reflects accumulated evidence of point k being an exemplar based on the received positive responsibilities from other points.

All availabilities are initialized to zero: $a(i,k) = 0$. After the updating converges, availabilities and responsibilities are combined to identify exemplars. For point i, its corresponding exemplar is obtained by maximizing the following expression:

$$k^* = \arg\max_k \{a(i,k) + r(i,k)\}. \tag{6}$$

We choose AP for the following reasons.

1) AP can find clusters with much lower error than other clustering methods, such as K-Means [9].
2) AP performs efficiently on sparse similarity graphs, which is the case of document space. The run time for iterations is linear with the number of edges in the graph.
3) AP takes a real number as input, called the preference for each data point. The preference quantifies the likelihood of it being chosen as an exemplar. Thus, prior and heuristic knowledge can be used to associate different sentences with different preferences.
4) AP identifies exemplars for each cluster or group that can be naturally used as the summary sentences for the cluster.

2) Hierarchical Summarization on Affinity Propagation: For the sentences in related reports,$\{s_1, s_2, \ldots, s_n\}$, we want to build a hierarchical clustering structure and use exemplars of clusters as the summary. Starting from all sentences, we recursively apply AP in an agglomerative way to find proper exemplars until the number of exemplars is small enough. We pick 20 as the number of exemplars which means 20 sentences will be selected from the document set as the summary. The preference for each sentence and similarity between sentences are used as the input of the AP algorithm.

3) Sentence Preference: We define the preference of sentence i to be chosen as an exemplar using the following scores.

1) $LanguageModelScoreL$: For sentence i, L_i is calculated as the logarithmic probability of sentence i using unigram model training on the reports $\{s_1, s_2, \ldots, s_n\}$. Generally, a shorter sentence that has more frequent words in the reports has a higher score.
2) *LexPageRankscore* P: LexPageRank proposed in [10] calculates the page rank score of sentences on the sentence similarity matrix. The score measures the prestige in sentence networks assuming that the sentences similar to many of the other sentences in a cluster are more prestigious with respect to the topic. Since the original LexPageRank can be interpreted as the probability in random walk theory, we use the logarithmic version to make it at the same scale with the language model score.
3) *FreshnessScore* F: Users are generally more interested in the latest information; we calculate the freshness score of sentence i based on the age of the document containing i as

$$F_i = e^{-a_i} \tag{7}$$

where a_i is the *age* in terms of the number of days the document contains the sentence i. Clearly, $F_i \in [0,1]$ decreases as the document age increases. Another property is that for two sentences from two documents with some age difference (e.g., 1 day), the difference of their freshness scores is large when both sentences are relatively new. Thus, it can better differentiate freshness for latest information.

Finally, the preference of s_i is the sum of the three feature scores with a scaling parameter:

$$s(i,i) = (L_i + D_i + F_i) \times e. \tag{8}$$

The parameter e is obtained by experimentally testing the clustering results and choosing the value that achieves the best clustering performance.

4) Sentence Similarity: Sentence similarity $s(i,j)$ indicates how well the data point with index j is suited to be the exemplar for data point i. In our case, it means how likely sentence i can be summarized by sentence j. If sentence i and sentence j have nonstop word overlaps, we calculate $s(i,j)$ by the log-likelihood of sentence i given that its exemplar is sentence j as follows:

$$s(i,j) = \log P(i|j). \tag{9}$$

To calculate the conditional probability, a unigram language model is trained on sentence j by using the Dirichlet smoothing [32]. Then, the probability of sentence i is calculated by using the language model.

C. Dynamic Dashboards and Dynamic Query Form

1) Dynamic Dashboard:

a) Challenges for dashboards: When a disaster happens, the system will receive a lot of information at once. It is necessary for the system to select a small portion of entities that a user really cares about to display in the dashboards. The dashboards provide condensed views for users to quickly explore the recent news and reports. It cannot display all the information in such a small area.

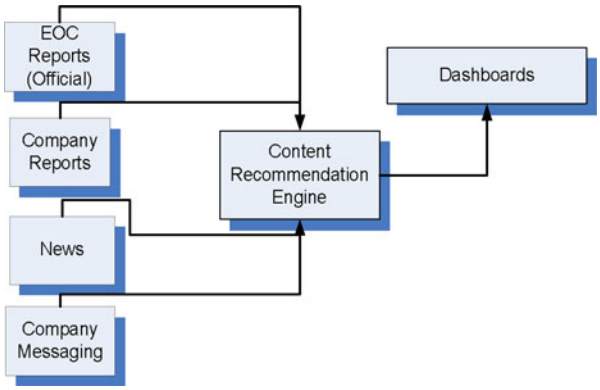


Fig. 1. Content recommendation engine.

Another problem in practice is that the information sent from company users may have a lot of redundancy. For instance, when a hurricane arrives in South Florida, almost all the company users in that area will report the same hurricane information: "*The storm has arrived South Florida*." Thus, different users may report the same information hundreds of times. Therefore, the system has to identify which information is redundant, and the redundant information should not appear in the dashboards.

We address these problems by introducing the dynamic dashboard supported by the content recommendation engine. The engine's main task is to extract the most important, relevant and nonredundant information about entities from news and reports.

b) Content recommendation engine: Fig. 1 shows the data flow related to the content recommendation engine. There are four main data sources: EOC reports, news, company reports, and company messages. Since reports and news may contain information about multiple entities, in content recommendation engine, each report or news is divided into several documents. Each document consists of a sentence that contain entity status information plus a context window (one previous and next sentence).

The content recommendation consists of two steps. The first step is text clustering, which is to cluster the same description of entities into one cluster. The second step is ranking the text by the relevance and presenting the top k items to the dashboards.

The content recommendation engine is based on unstructured text, while the situation dashboard, thread dashboard, and company dashboard display structured information. The four dashboards are denoted as Db_S(situation dashboard), Db_T(threat dashboard), Db_E(event dashboard), and Db_C(company dashboard). The maximum numbers of items allowed to show in the dashboards Db_S, Db_T, Db_E, and Db_C are denoted as size_S, size_T, size_E, size_C, respectively.

The content recommendation engine recommends information from different data sources to the four dashboards. Table V shows the relationship between the data sources and the four dashboards. Since the dashboards show the latest information, we use the last 48 h records and news as the input of the engine.

For any user u, the set of information submitted by u is denoted by $I(u)$ and the set of reports/news of which the details are viewed by u is denoted by *J(u)*. *u*'s profile is composed of *I(u)* and *J(u)*.

TABLE V
DATA SOURCES OF DIFFERENT DASHBOARD

Dash-board	EOC Reports	Company Reports	News	Company Messaging
Db_S	√		√	
Db_T	√			
Db_F	√	√	√	√
Db_C		√		√

c) Document clustering: Before performing clustering, we use term frequency–inverse document frequency (TF–IDF) transformation [11] to transform the text data (report, news, and so on) to the vectors. The similarity between two documents can be calculated by the cosine similarity [12].

We apply the K-Medoids [13] algorithm to cluster the documents. Note k is a user-defined parameter, which is determined by the managers of the system. It is also relevant to the number of items allowed to be displayed on the dashboards. We present the top five $(k = 5)$ items in the dashboards.

After clustering, each cluster contains the duplicated information about an entity and one document can be selected from a cluster to show the status of the entity. However, before that, we have to decide which cluster and which document should be selected.

d) Content ranking: For a specific user u, there are three priorities of the information. The three priorities from highest to lowest are EOC reports, company partner's information (messages received) and other users' information (company reports). The three priorities are denoted by user-defined parameters pr_1, pr_2, and pr_3, respectively, and $pr_1 > pr_2 > pr_3 > 0$. For a given document $d_i \in D$, we use $pr(d_i)$ to indicate the priority of this document, and $pr(d_i) \in \{pr_1, pr_2, pr_3\}$.

Suppose the current user is u, $t(u)$ represents the term vector representation of the documents submitted or read by *u* .We can obtain the u's feature f_u by users' profiles as follows:

$$f_u = \alpha \frac{\sum_{u \in I(u)} t(u)}{\left|\sum_{u \in I(u)} t(u)\right|} + (1-\alpha) \frac{\sum_{u \in J(u)} t(u)}{\left|\sum_{u \in J(u)} t(u)\right|}. \quad (10)$$

The parameter α is used to tune the importance weights of the reports submitted and viewed as the profile. α is set to 0.8 in our work.

The importance score of each document $d_i \in D$ is calculated as follows where $t(d_i)$ represents the term vector representation of document d_i:

$$\mathrm{score}(d_i) = \mathrm{sim}(f_u, t(d_i)) \cdot pr(d_i). \quad (11)$$

For each dashboard, we use a top-K query to greedily search the K highest scores' documents from its corresponding data sources, where $K \in \{\mathrm{size}_S, \mathrm{size}_T, \mathrm{size}_E, \mathrm{size}_C\}$ and no two documents are selected from the same cluster. The set of K highest scores' documents is the result of the content recommendation engine. The EOC official reports have the highest priority. Some of them are not very relevant to the current user; however, information from these reports is still likely to appear on the event dashboard.

2) Dynamic Query Form: Each report is associated with a set of attributes, such as the report location, date, or annotations added by the creator. Such structural information allows users to execute relational queries on reports. For example, we want to find those reports that are about hurricanes from 1990 to 2010 and the latitude of the hurricane center is above 30°. Hence, our system applies query forms for users to support relational queries.

Traditional query forms are statically embedded by developers or database administrators. Those static query forms are used for the static database schema. However, different reports have different sets of attributes. For example, the hurricane report and the earthquake report use two very distinct sets of attributes. Furthermore, the associated values of annotation attributes created by the user at runtime are inconsistent. Therefore, it is impossible to design a static and fixed query form to cover all those attributes. Therefore, we implement the dynamic query form to satisfy those dynamic and heterogeneous query desires.

Previous research on database query forms focuses on how to automatically generate the query form from the data distribution or query history [14]–[17]. However, different users can have different query desires. How to capture the current user's interests and construct appropriate query forms are the key challenges for query form generation that has not been solved.

a) Problem formulation: Query forms are designed to return the user's desired results. The metric of the goodness of a query form is based on two traditional measures of evaluating the quality of the query results: *precision* and *recall*.

Let $F = (\mathbf{A}_F, \sigma_F)$ be a query form with a set of query conditions σ_F and a set of displaying attributes $\mathbf{A}_F$. Let D be the set of all reports in the database; $|D|$ is the total number of reports. $P_u(.)$ is the distribution function of user interests; $P_u(d)$ is the user interest for a report d, and $P_u(A_F)$ is the user interest for an attribute subset A_F; and $P(\sigma_F|d)$ is the probability of query condition σ_F being satisfied by d, i.e., $P(\sigma_F|d) = 1$ if d is returned by F and $P(\sigma_F|d) = 0$ otherwise. Then, given a query form $F = (\mathbf{A}_F, \sigma_F)$, the *expected precision*, *expected recall*, and *expected fscore* of F are defined as follows:

$$\text{Precision}_E(F) = \frac{\sum_{d\in D} P_u(d)\, P_u(A_F)\, P(\sigma_F|d)}{\sum_{d\in D} P(\sigma_F|d)} \tag{12}$$

$$\text{Recall}_E(F) = \frac{\sum_{d\in D} P_u(d)\, P_u(A_F)\, P(\sigma_F|d)}{\sum_{d\in D} P_u(d)\, P_u(A)} \tag{13}$$

$$F\text{Score}_E(F) = \frac{\left(1+\beta^2\right)\cdot \text{Precision}_E(F)\cdot \text{Recall}_E(F)}{\beta^2\cdot \text{Precision}_E(F)\cdot \text{Recall}_E(F)} \tag{14}$$

where $A_F \subseteq A, \sigma_F \in \sigma$, and β is a parameter defined by the user and β is usually set to 2.

$F\text{Score}_E(.)$ is the metric to evaluate the overall goodness of a query form. The problem of our dynamic query form [43] is how to construct a query form $\hat{F}$ that maximizes the goodness metric $F\text{Score}_E(.)$, i.e.

$$\hat{F} = \underset{F}{\text{argmax}}\, F\text{Score}(F). \tag{15}$$

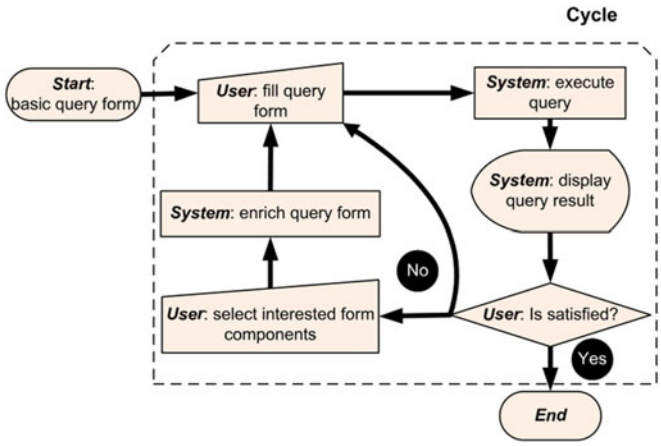


Fig. 2. Flowchart of dynamic query form.

b) Method description: It is impractical to construct an optimal query form $\hat{F}$ at the very beginning, since we do not know which reports and attributes are desired by the user. In other words, estimating $P_u(d)$ and $P_u(\mathbf{A}_F)$ is difficult.

The ADSB system provides an iterative way for the user to interactively enrich the query form. Fig. 2 shows the work flow of our dynamic query form system. At each iteration, ADSB computes a ranked list of query form components for users, and then, lets users make the choice for their query form. Those query form components are ranked by the metric $F_E(F)$.

There are two types of query form components: attribute display and query condition.

Assuming the current query form is F_i in the flowchart, and the next query form is F_{i+1}. We need to estimate $P_u(d)$, $P_u(\mathbf{A}_{F+1})$, and $P(\sigma_{F+1}|d)$ to compute $F\text{Score}_E(F_{i+1})$. The estimation is based on user behaviors when interacting with the ADSB system. Let $D_{u,f}$ be the set of reports viewed by the users and d' be one of the document in $D_{u,f}$. We assume those reports are interesting to the current user, then

$$P_u(d) = \sum_{d'\in D_{u,f}} P_u(d|d')\, P_u(d'). \tag{16}$$

We use the random walk model to compute the relevance score between reports as the value of $P_u(d/d')$ [18].

Suppose A is displaying an attribute we suggest for query form F_{i+1} and $\boldsymbol{A}_{f_{i+1}} = A \cup \boldsymbol{A}_{F_i}$, where $A \in \boldsymbol{A}, A \notin \boldsymbol{A}_{F_i}$. Therefore, $\boldsymbol{A}_{F_i}$ can be obtained in the current query form F_i.

$$P_u\left(\boldsymbol{A}_{F_{i+1}}\right) = P_u\left(A|\boldsymbol{A}_{F_i}\right) P_u\left(\boldsymbol{A}_{F_i}\right). \tag{17}$$

We also estimate $P_u\left(A|\boldsymbol{A}_{F_i}\right)$ by using a random walk model on the *attribute graph*. The nodes of the attribute graph are report attributes, and the edges are common reports. Therefore, the weight of edge *ij* is computed by how many reports use both the two attributes i and j.

Suppose s is a query condition we suggest for query form F_{i+1}. Therefore, $\sigma_{F_{i+1}} = s \wedge \sigma_{F_i}$, where s is a single query condition for attribute $A_s, A_s \in \boldsymbol{A}$. σ_{F_i} can be obtained in the current query form F_i. For each report $d \in D, P\left(\sigma_{F_{i+1}}|d\right) = P(s|d)\, P\left(\sigma_{F_i}|d\right)$. It is very time consuming to find the best s

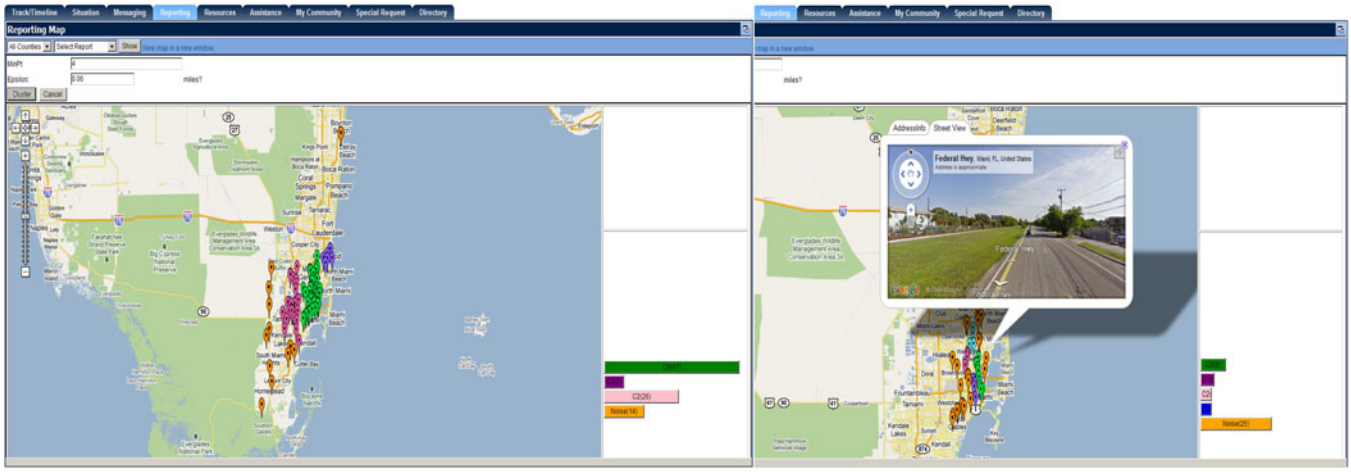


Fig. 3. Dynamic community generation result. (a) Generated communities. (b) Interactive clustering of large cluster.

by brute-force search on all $P(s|d)$. Therefore, we precompute the $P(s|d)$ and store it in the database.

D. Community Generation and User Recommendation

1) Community Generation: Two characteristics in disaster recovery scenarios motivate us to consider geo-location information. The first characteristic is that any event extracted from a report is associated with a/several location(s) indicating the place(s) where the announced event takes place. The second characteristic is that spatially colocated entities are more likely sharing similar disaster damage situations.

These two characteristics motivate the concept of community: a community is a certain geographical region in which entities tend to share more recovery status or interests in common. Therefore, geographically identifying those communities is important to help companies understand the current disaster situation and any interested resources nearby. Our system addresses community generation by adapting existing spatial clustering algorithms. In practice, we provide an interactive spatial clustering interface for users to access multilevel communities in a top–down manner and consider physical or nonphysical obstacles when generating spatial clusters to form more practical communities.

a) Spatial clustering: Spatial data clustering identifies clusters, or densely populated regions, according to some distance measurement in a large, multidimensional dataset [12], [13]. Many spatial clustering techniques [19], [20], [26] have been developed to identify clusters with arbitrary shapes of various densities and with different physical constraints.

In practice, communities formed by geographically related entities can be of various shapes. Therefore, we extend DBSCAN [19], a well-known density-based clustering algorithm, which is capable of identifying arbitrary shape of clusters, to generate dynamic communities.

b) Spatial clustering with constraints: We consider the method of spatial clustering with constraints. Generally, there are three types of constraints [13]. 1) Constraints on individual objects: such constraints are nonspatial instance-level constraints that can be preprocessed before performing clustering algorithms. 2) Constraints as clustering parameters: such constraints are usually confined to the algorithm itself. Usually, user-specified parameters are given through empirical studies. 3) Constraints as physical obstacles: such constraints are tightly intertwined with clustering process. It is clear that physical obstacles are such constraints that prevent two geographically close entities from being clustered together. For example, the bridge, highway and rivers are of this type.

In our BCIN system, we focus on object constraints and physical constraints.

Object constraints: We have two ways to obtain object constraints: 1) users submit formatted reports through report interface. Those reports are immediately recorded in the database; 2) our system extracts entity status from reports. For example, Table IV can be used as object constraints.

Obstacle constraints: A polygon is a typical structure in spatial analysis for modeling objects. Obstacles modeled by a polygon can be represented as a set of line segments after performing polygon reduction [20].

Fig. 3(a) shows the communities generated by clustering all open facilities and companies in Miami with the constraint: "I75 closed."

c) Interactive spatial clusters: In order to deal with unbalanced size of clusters, we provide users with an interactive mechanism to track the subcommunity information within a large size community. Further clustering process will be triggered in the runtime when a user selects a larger community and wants to see the cluster information within such a community at a finer granularity. By using this mechanism, users can obtain clusters with different granularities and more meaningful results. Fig. 3(b) shows the interactive clustering results within the largest cluster in Fig. 3(a).

2) User Recommendation: The user recommendation component provides an interface to explore other users' recommendations or share reports with other people. It also helps the user quickly identify sets of users with shared interests. It is designed

by considering each individual's transactional sharing history, textual content of each transaction and timeliness of interaction to provide each user with a personalized information sharing experience.

Related work has been applied to email communication networks analysis to find important persons, identifying frequent communication pattern and detecting communities based on transactional user relationships [21]–[25]. Those techniques can prevent a user from forgetting to add important recipients, avoid costly misunderstandings, and communication delays. Carvalho *et al*. [24] introduced several supervised learning models to predict the score of each user associated with a given email content. By aggregating TF–IDF vector of each email addressed to a user (by To, CC, or BCC), it can predict the score of a new email to such user. However, it was not aware of the different importance of emails for senders and recipients. Horn *et al*. [25] explicitly associated higher weights to senders, and also consider user-interaction graph as a directed hypergraph. It focused on the time and frequency of interactions but ignored the content information involved in each email, which could be an important indication of potential related users.

There are three practical considerations motivating the user recommendation: 1) to share information to the right/related people, users need an intelligent tool to autogenerate a recipient list that covers active users interested in specific information; 2) manually identifying meaningful groups of users is time consuming, therefore, users prefer efficient ways to organize contacts instead of navigating the contact list repeatedly; and 3) it could be more effective for a user to access information that others think is important.

Therefore, our system addresses the aforementioned issues by considering both user interactions and textual information. In practice, we provide dynamic user suggestions for the news recommendation and community recommendation interface to help our system users organize their critical partnerships.

a) Transactional interactions: An interaction or transaction is defined as the process of a user sharing a report with one or more other users. Therefore, the reports sharing transaction database can be treated as a hypergraph with each node representing a registered user and a set of edges created at the same time from one node to a set of nodes representing an occurred transaction. There are three important factors associated with each edge.

1) *Time:* The time that the transaction happened. It indicates the importance of recency. In general, the more recently a transaction happens, the more important the report is to those users involved.

2) *Direction:* The relation of an interaction. An edge pointed from node A to node B, which indicates that A shares some information with a set of users including B. The direction indicates that the shared information is more important to the sender than to receivers.

3) *Textual Content:* Each transaction is associated with some specific textual content, so the content of an edge means that someone thinks such content is important or related to some group of users.

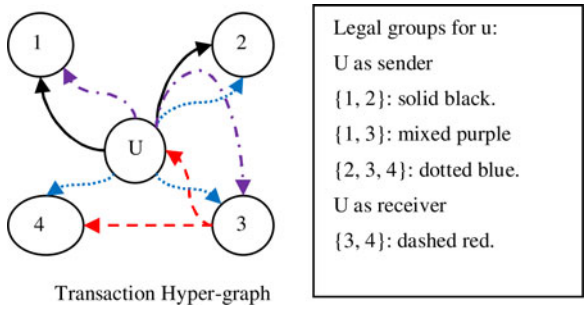


Fig. 4. Transactional user groups.

In practice, a personalized user recommendation requires the algorithm to identify potential users who have frequent and active interactions with the sender and are also interested in specific topics. In completion of two recommendation tasks, we extend both [24] and [25] by taking the direction, timeliness and textual content of the interaction into consideration to generate: 1) a suggested user list for specific report and 2) a suggested user list for specified seeds (users).

b) User groups: There could be multiple transactions associated with a specified user and each transaction involves a group of users (see Fig. 4).

Even though transactions may include the same sender and receivers, they are treated as unique in the transactional hypergraph since they are associated with unique timestamps. Despite the textual content of each transaction, the contribution of each group to current user's seeds can be evaluated by interaction rank proposed in [25].

c) User profile: To build the user profile, we consider textual content in all transactions related to the user. Carvalho [24] introduced a centroid vector-based representation, which aggregates all related documents to build a user profile. In our method, we consider transaction directions and assign document sending weight $\mathcal{W}_s$ or receiving weight $\mathcal{W}_r$ respectively. The values of $\mathcal{W}_s$ and $\mathcal{W}_r$ are manually decided based on different scenarios. For example, if the relevance of an email to the sender is higher than to the receiver (for example, junk mail or ads), then we can assign a much larger value to $\mathcal{W}_s$ than to$\mathcal{W}_r$. We use TF–IDF transformation to represent textual content as a vector. Therefore, the user profile can be represented as

$$\text{profile}\,(u) = \mathcal{W}_s \cdot \sum_{d \in S(u)} \text{tfidf}\,(d) + \mathcal{W}_r \cdot \sum_{d \in R(u)} \text{tfidf}\,(d) \tag{18}$$

where $\text{tfidf}\,(d)$ is defined as

$$\text{tfidf}\,(d)_i = \text{TFIDF}\,(d)_i^t \tag{19}$$

where $t = \frac{\text{time(now)} - \text{time}(n)}{\lambda}$ indicates an over-time exponential decay of each document's contribution. $S\,(u)\,, R\,(u)$ are sets of documents sent and received by u, respectively. Therefore, user u's preference to report d can be generated by computing the cosine similarity between the user's profile and the TF–IDF vector of d

$$\text{preference}\,(u, d) = \cos\left(\text{profile}\,(u)\,, ts_{\text{tfidf}}\,(d)\right). \tag{20}$$

Input: u, the user; d, the report, and $\mathcal{S}$, the seeds
Output: $\mathcal{R}$, recommended user list
1. $\mathcal{G} \leftarrow$ *GetTransactionalGroups(u)*
2. $\mathcal{R} \leftarrow \emptyset$
3. for each group $g \in \mathcal{G}$
4. for each user $c \in$ g, $c \notin \mathcal{S}$
5. if $c \notin \mathcal{R}$
6. $\mathcal{R}[c] \leftarrow 0$
7. $\mathcal{R}[c] \leftarrow \mathcal{R}[c]$ + *GroupScore*(c, $\mathcal{S}$, g, d)
or $[c] \leftarrow \mathcal{R}[c]$ + *CommunityScore*(c, $\mathcal{S}$, g)

Fig. 5. Suggesting user routine.

Practically, the user profile is stored separately and will not be updated in each calculation. Typically, it will be updated every few days or when new events are announced.

d) User group suggestion algorithm: We extended the friend-finding algorithm proposed in [25] to generate a list of user recommendations by aggregating the groups' contribution to a user and considering the relevance between users and reports. Our algorithm is described in Fig. 5. The score of each user in the list represents the interaction preference with respect to the given user and report.

e) Group contribution: From the algorithm described in Fig. 5, the interaction preference of a user is the aggregated value of the contribution that each transaction made to the user. There are two types of contribution measurements with respect to different tasks. We use group score and community score to represent contributions for report sharing and community user recommendation, respectively.

f) Group score: The group contribution GC described later represents the contribution that a user group contributes to a user. There are two situations considered: 1) suggesting users related to a document based on the preference (similarity) between the document and a user; and 2) suggesting a user group based on the similarity between users. We defined GC as an aggregated score of users' preferences to a specific document considering the direction and timeliness of each interaction.

For the first situation, we use similarities between each user in a group and report d:

$$\mathcal{GC}(d,g) = \mathcal{W}_s \cdot \sum_{i \in O(u,g)} s(i,d)^t + \mathcal{W}_r \cdot \sum_{i \in I(u,g)} s(i,d)^t \tag{21}$$

where $s(i,d) = \sum_{u \in i} \text{preference}(u,d)$.

For the second situation, we simply modified the $\mathcal{GC}(d,g)$ as $\mathcal{GC}(c,g)$ and $s(i,d)$ as

$$s(i,c) = \sum_{u \in i} \cos(\text{profile}(u), \text{profile}(c)) \tag{22}$$

to calculate the similarity without document information.

In both situations, $O(u,g)$ and $I(u,g)$ are sets of sending and receiving interactions/transactions, respectively, in which user u was involved.

g) Recommend users with report: To recommend a report to a group of users, one should consider historic recommendation transactions and the report's textual content. The score that a transaction contributes to a user is the aggregation of preferences of a group of users to the given report

$$\text{GroupScore}(c,\mathcal{S},g,d) = \begin{cases} \mathcal{GC}(d,g), & \text{if } S \cap g \neq \emptyset; \\ 0, & \text{otherwise.} \end{cases} \tag{23}$$

h) Recommend users for communities: Recommending users to form communities involves historic transactions without textual information. The score that a transaction contributes to a user is the aggregation of similarities between the user and users in the group

$$\text{CommunityScore}(c,\mathcal{S},g) = \begin{cases} \mathcal{GC}(c,g), & \text{if } S \cap g \neq \emptyset; \\ 0, & \text{otherwise.} \end{cases} \tag{24}$$

A user can arbitrarily choose target users at runtime. Starting from those chosen users as seeds, our recommendation components can dynamically generate more users related to the given textual content and list of users with high concurrence.

III. Case Study

A. Business Continuity Information Network

The BCiN (see Fig. 6) is a web-based prototype implementation of a Business Continuity Information Network that is able to link participating companies into a community network, provide businesses with effective and timely disaster recovery information, and facilitate collaboration and information exchange with other businesses and government agencies. The system allows company users to submit reports related to their own business, and government users to make announcements on issues impacting the public. To collect more information during the disaster, BCiN can monitor the news published on the websites and takes the news as its input. Like traditional information systems, these reports and news, and the status information of entities they contain can be retrieved and accessed by queries. For example, reports can be viewed according to alert categories or geo-locations, and resources can be viewed according to status or usages. Furthermore, BCiN not only displays user-submitted information but also conducts necessary and meaningful data processing work. BCiN makes recommendations based on the current focus and dynamically adapts based on users' interests. BCiN summarizes reports and news to provide users with brief and content-oriented stories, which prevent users from being troubled when searching through large amounts of information. By introducing the concept of community, BCIN offers users a hierarchical view of important reports or events around them.

Four main information processing and representation components are implemented in BCiN: information extraction (see Section II.A), report summarization (see Section II.B), dynamic dashboard (see Section II.C.1), and dynamic community generation (see Section II.D.1). These four different components are tightly integrated to provide a cohesive set of services and constitute a holistic effort on developing a data-driven solution for disaster management and recovery.

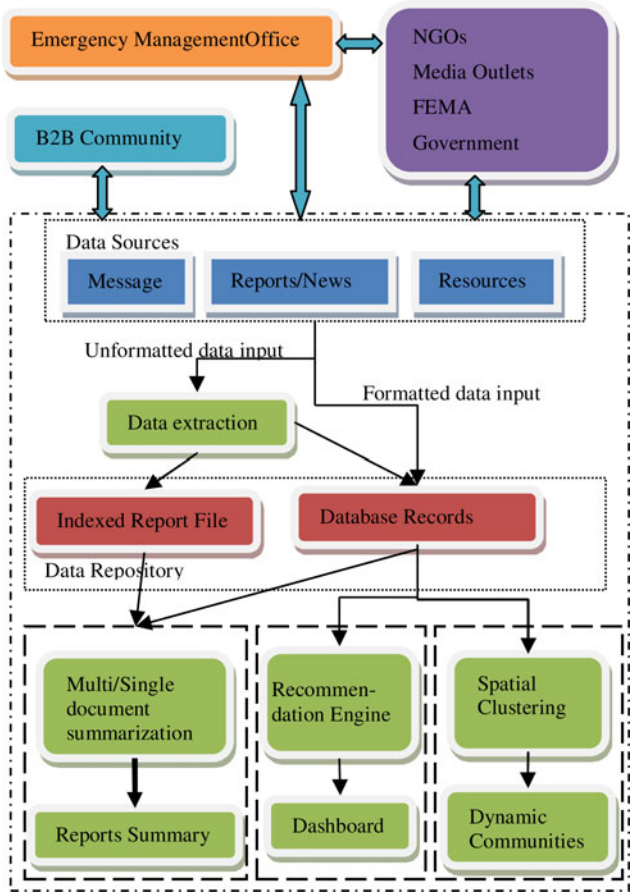


Fig. 6. BCIN system architecture.

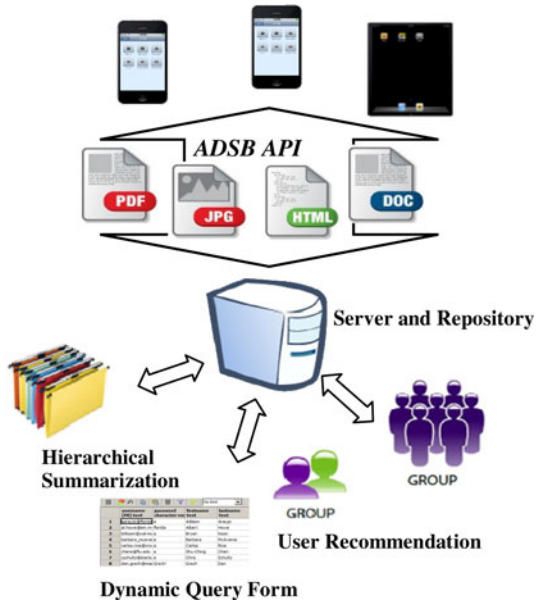


Fig. 7. ADSB system architecture.

B. All-Hazard Disaster Situation Browser

Professionals who have an operational responsibility in disaster situations are relying on mobile phones to maintain communications, update status, and share situational information. Consumers, too, are finding mobile devices convenient for sharing information about themselves and what is going on in their lives. By using a mobile platform, we can build native applications that utilize onboard sensors, rich media, and simplified user interfaces to engage users in a way that they feel is most comfortable for sharing such information in a disaster situation.

ADSB is an *All-Hazard Disaster Situation Browser (ADSB)* system that runs on Apple's mobile operating system (iOS), and iPhone and iPad mobile devices. Fig. 7 illustrates the system architecture, and Fig. 8 illustrates the system screenshot. Four major components are implemented in ADSB: information extraction (see Section II.A), hierarchical summarization (see Section II.B), dynamic query form (see Section II.C.2), and user recommendation (see Section II.D.2). A video demonstration is available at http://users.cis.fiu.edu/∼taoli/ADSB-Demo/demo.htm.

IV. System Evaluation

The data sources used in our project can be broadly divided into two categories based on temporal characteristics: static data sources and dynamic data sources. Static data sources include historical data from Miami-Dade EOC. Dynamic data sources include: 1) situation reports from Miami-Dade EOC and participating companies illustrating the current status of threat, ongoing operations, and goals/objectives for preparation and recovery efforts; 2) open/closure status about roadways/highways/bridges and other infrastructure such as fuel, power, transportation, emergency services (fire stations, police stations), schools, and hospitals; 3) reports crawled from FEMA [28] web site with information about 20 major disasters since 2000; and 4) tweets posted in August 2010 by using Twitter API [27] from dozens of active accounts.

Evaluation is conducted on two levels: algorithm evaluation and system evaluation. To evaluate the algorithms, we use standard performance metrics and compared our algorithms with existing work when applicable. Using report summarization as an example, we conducted experiments on a dataset of press releases collected from Miami-Dade EOC and Homeland Security during Hurricane Wilma from October 19, 2005 to November 4, 2005. The dataset contains 1700 documents in total, concerning all the related events before Hurricane Wilma came, during Hurricane Wilma, and after Hurricane Wilma passed [42]. The documents report various types of information such as the movement of Hurricane Wilma, the location of evacuation zones, and the cancellation of social activities. In order to evaluate the summarization performance, human generated summaries are used as references. The summarization results are evaluated by ROUGE [33].

Table VI shows the experimental results and demonstrates the efficiency of using AP to generate hierarchical document summarization (Centroid means picking the cluster centroid as the representative sentence).

Our system evaluation process consists of presenting the system to emergency managers, business continuity professionals, and other stakeholders for feedback and performing community exercises. The exercises involve a real-time simulation of

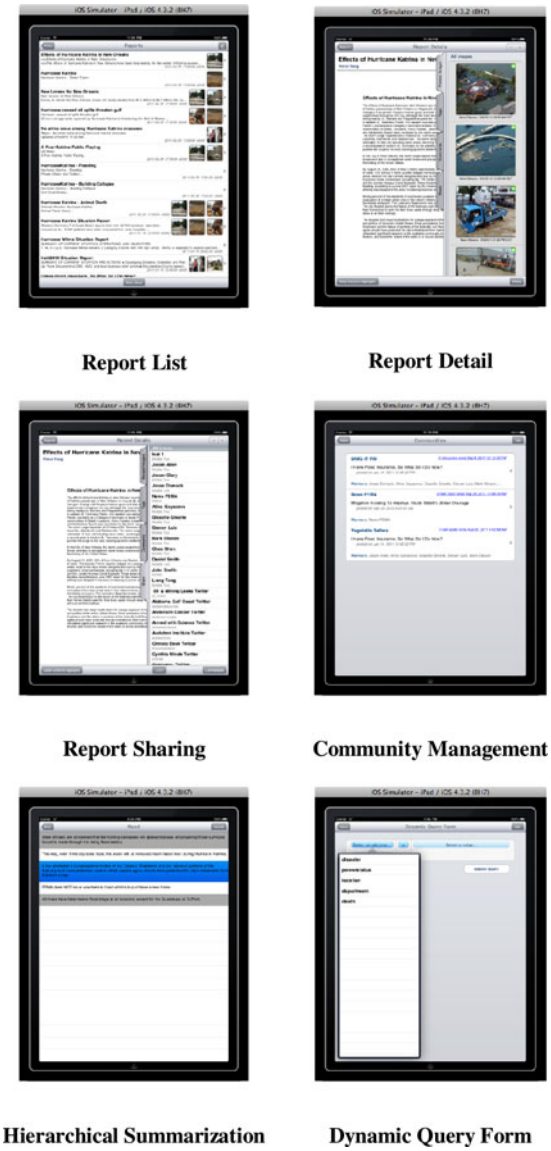


Fig. 8. ADSB screen shots of important components. *iPhone implementation has the same style with iPad but without rich visual abilities, such as the split view.

TABLE VI
SUMMARIZATION RESULTS COMPARISON

Measure		Centroid	Affinity Propagation
ROUGE-1	Recall	0.3409	**0.3788**
	Precision	0.1991	**0.3311**
	F-Score	0.2514	**0.3534**
ROUGE-2	Recall	0.0916	**0.1069**
	Precision	0.0533	**0.0933**
	F-Score	0.0674	**0.0996**
ROUGE-SU4	Recall	0.1121	**0.1173**
	Precision	0.0649	**0.1023**
	F-Score	0.0822	**0.1093**

TABLE VII
EVALUATION EXERCISES

Date	Description of the Exercise
Jun. 01 2009	In Florida Dept. of Emergency Management's Statewide Hurricane Exercises, BCiN was utilized in a scenario where Miami-Dade County Emergency Management Business Recovery Desk vafacilitated the logistics to deploy portable ATMs at Shelters and PODs in Miami-Dade County.
Jun. 29 2009	In Miami-Dade UASI exercise, BCiN supported communicating and collaborating with several companies that participated in the event as observers.
Aug. 20 2009	In a full scale company BCiN training, about 30 companies were given injects to provide information to resolve different information requests.
May 10 2010	In Miami-Dade Dept. of Emergency Management's Statewide Hurricane Exercise, our systems were responsible for disseminating and responding to injects during the course of the exercise for both government and company users.
Jul. 29 2010	In Miami-Dade company exercises, over 50 company attendees used our systems for a training exercise.
May 12 2011	In the county of West Palm Beach exercise, we demonstrated the system to WPB Dept. of Emergency Management and companies.

a disaster event integrated into an existing readiness exercise conducted each year. This evaluation exposes information at different time intervals and asks the community to resolve different scenarios by using the tool. The evaluation is a form of a "table-top" exercise in which injected information provides details about the current disaster situation and specifies potential goals and courses of action. Participant use the system to gather information to assess the situation and provide details about the actions they will take. We gather information about what information they found to derive their conclusions (or lack thereof). This information allows us to better understand how those techniques improve the information effectiveness.

Table VII describes the exercises. In a regional disaster such as a hurricane, business continuity professionals are under extreme pressure to execute their continuity of operation plans because many of the usual sources of information and services about the community and supply chain are completely disconnected, sporadic, redundant, and many times lack actionable value. The system focuses user input and collaboration around actionable information that both public and private sector can use.

To validate the usability and performance of our system, the participants and the EOC personnel at Miami-Dade participated in the questionnaire session after the exercise. A set of ten questions was designed to evaluate our system where nine of them are multiple choice questions with a five-level scale (strongly agree, agree, not sure, disagree, and strongly disagree) and the last one is an open-ended question. Some of the multiple choice questions are: Are you able to identify related reports that you are interested in? Are you able to identify the correct modules for your tasks? Are you able to switch between different modules? Are the system generated summaries useful? The open-ended question is about feedback and suggestions from the users. On average, about four EOC personnel and 30 participants attended each exercise. The evaluation demonstrated that most of participants are satisfied with the performance of the tools. Specifically, seven out of nine multiple choice questions received "strongly agree" or "agree" from over 90% of the participants, implying a high level of satisfaction with our system.

The feedback from our users is positive and suggests that our system can be used not only to share the valuable actionable information but to pursue more complex tasks like business planning and decision making. There are also many collaborative missions that can be undertaken on our system, which allows public and private sector entities to leverage their local capacity to serve the recovery of the community. We summarized the feedback as follows.

1) *Positive feedback:* a) the system is easy to use; b) related reports are well organized based on personalized user groups; and c) reports summarization is representative and interesting.
2) *Some suggestions:* a) related multimedia information, including images and video, could be shown during navigation; b) report summaries could be organized based on some points of interests.

V. Conclusion

We identified four key design challenges to support multiparty coordination during disaster situations. We proposed a unified framework that systematically integrates the different techniques that are developed in our previous work [5], [29]. Such a framework can be utilized when dealing with different systems or applications separately (e.g., BCiN and ADSB), and they are essentially collaborative platforms for preparedness and recovery that helps disaster impacted communities to better understand what the current disaster situation is and how the community is recovering. The system evaluation results demonstrate the effectiveness and efficiency of our proposed approaches.

During the system implementation and assessment process, the users provided suggestions, limitations and possible enhancements. Our future efforts will be focusing on the following tasks: developing efficient tools to automatically crawl related information from public resources including news portals, blogs, and social Medias; capturing the current user's interests and construct appropriate query form; and understanding users' intends to provide them with actionable answers to their information inquiries.

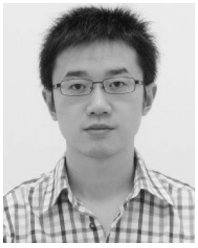
Acknowledgment

The authors would like to thank J. Domack, M. Oleson, and J. Allen for their work in the system development and testing. The initial work has been recognized by FEMA (Federal Emergency Management Agency) Private Sector Office as a model in assistance of Public-Private Partnerships [2].

References

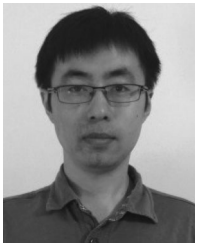
[1] H. Muson, "Preparing for the worst: A guide to business continuity planning for mid-markets," Executive Action Series, The Conference Board, Rep. A-0179-06-EA, Feb. 2006.
[2] FEMA public Private Partnership Models. [Online]. Available: http://www.fema.gov/pri-vatesector/ppp_models.shtmunder Miami-Dade County.
[3] K. Saleem, S. Luis, Y. Deng, S.-C. Chen, V. Hristidis, and T. Li, "Towards a business continuity information network for rapid disaster recovery," in *Proc. Int. Digit. Gov. Res. Conf.*, 2008, pp. 107–116.
[4] V. Hristidis, S. Chen, T. Li, S. Luis, and Y. Deng, "Survey of data management and analysis in disaster situations," *J. Syst. Softw.*, vol. 83, pp. 1701–1714, 2010.
[5] L. Zheng, C. Shen, L. Tang, T. Li, S. Luis, S. Chen, and V. Hristidis, "Using data mining techniques to address critical information exchange needs in disaster affected public-private networks," in *Proc.16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2010, vol. 10, pp. 125–134.
[6] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. 18th Int. Conf. Mach. Learning*, 2001, pp. 282–289.
[7] F. Sha and F. Pereira, "Shallow parsing with conditional random fields," in *Proc. HLT-NAACL*, 2003, pp. 131–141.
[8] C. W. Hsu and C. J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, Mar. 2002.
[9] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 21, pp. 972–976, 2007.
[10] G. Erkan and D. R. Radev, "Lexpagerank: Prestige in multi-document text summarization," in *Proc. Empirical Methods Natural Language Process.*, 2004, pp. 365–371.
[11] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, 1983.
[12] P. N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*. Reading, MA, USA: Addison-Wesley, 2005.
[13] J. Han and M. Kamber, *Data Mining Concepts and Techniques*, 2nd ed. San Mateo, CA, USA: Morgan Kaufmann.
[14] M. Jayapandian and H. V. Jagadish, "Automated creation of a forms-based database query interface," in *Proc. VLDB*, 2008, pp. 695–709.
[15] M. Jayapandian and H. V. Jagadish, "Expressive query specification through form customization," in *Proc. 11th Int. Conf. Extending Database Technol.*, 2008, pp. 416–427.
[16] M. Jayapandian and H. V. Jagadish, "Automating the design and construction of query forms," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 10, pp. 1389–1402, Oct. 2009.
[17] P. P. Talukdar, M. Jacob, M. S. Mehmood, K. Crammer, Z. G. Ives, F. Pereira, and S. Guha, "Learning to create data-integrating queries," in *Proc. VLDB*, 2008, pp. 785–796.
[18] H. Tong, C. Faloutsos, and J. Pan, "Fast random walk with restart and its application," in *Proc. 6th Int. Conf. Data Mining*, 2006, pp. 613–622.
[19] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discoverying clusters in large databases with noise," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, 1996, pp. 226–231.
[20] C. H. Lee, "Density-based clustering of spatial data in the presence of physical constraints," Master's thesis, Univ. Alberta, Edmonton, AB, Canada, Jul. 2002.
[21] M. D. Choudhury, W. A. Mason, Jake M. Hofman, and Duncan J. Watts, "Inferring relevant social networks from interpersonal communication," in *Proc. 19th Int. Conf. World Wide Web*, Apr. 26–30, 2010, pp. 301–310.
[22] I. Kahanda and J. Neville, "Using transactional information to predict link strength in online social networks," in *Proc. 3rd Int. Conf. Weblogs Social Media*, Jun. 2009, pp. 74–81.
[23] S. Yoo, Y. Yang, F. Lin, and I. Moon, "Mining social networks for personalized email prioritization," presented at the 15th ACM SIGKDD Int. Conf. Knowledge Discovery Data Mining, Paris, France, Jun. 28–Jul. 01, 2009.
[24] V. R. Carvalho and W. W. Cohen, "Ranking users for intelligent message addressing," presented the 30th Eur. Conf. Advances Information Retrieval, Glasgow, U.K., Mar. 30–Apr. 03, 2008.
[25] I. Horn, A. Leichtberg, N. Leiser, Y. Matias, and R. Merom, "Suggesting friends using the implicit social graph," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, 2010, pp. 233–242.
[26] O. R. Aaiane, A. Foss, C. H. Lee, and W. Wang, "On data clustering analysis: Scalability, constraints, and validation," in *Proc. 6th Pacific-Asia Conf. Adv. Knowl. Discovery Data Mining*, 2002, pp. 28–39.
[27] Twitter API. [Online]. Available: http://apiwiki.twitter.com
[28] FEMA. [Online]. Available: http://www.fema.gov
[29] L. Zheng, C. Shen, L. Tang, T. Li, S. Luis, and S. Chen, "Applying data mining techniques to address disaster information management challenges on mobile devices," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, 2011, vol. 11, pp. 283–291.
[30] D. McEntire, "The status of emergency management theory: Issues, barriers and recommendations for improved scholarship," presented at the FEMA Higher Education Conf., Emmitsburg, MO, USA, 2004.
[31] GeoVISTA. [Online]. Available: http://www.geovista.psu.edu
[32] C. X. Zhai and J. Lafferty, "A study of smoothing methods for language models applied to ad hoc information retrieval," in *Proc. SIGIR*, 2001, pp. 334–342.
[33] C. Lin, "Rouge: A package for automatic evaluation of summaries," in *Proc. Workshop ACL*, 2004, pp. 25–26.

[34] E. Brill, "Part-of-speech tagging," in *Handbook of Natural Language Processing*. Boca Raton, FL, USA: CRC Press, 2000, pp. 403–414.
[35] The Puerto Rico Disaster Decision Support Tool (DDST). [Online]. Available: http://www.udel.edu/DRC/DDST/
[36] E. J. Bass, L. A. Baumgart, B. Philips, K. Kloesel, K. Dougherty, H. Rodríguez, W. Díaz, W. Donner, J. Santos, and M. Zink, "Incorporating emergency management needs in the development of weather radar networks," *J. Emergency Manage*., vol. 7, no. 1, pp. 45–52, 2009.
[37] L. A. Baumgart, E. J. Bass, B. Philips, and K. Kloesel, "Emergency management decision-making during severe weather," *Weather Forecasting*, vol. 23, no. 6, pp. 1268–1279, 2008.
[38] C. E. League, W. Díaz, B. Philips, E. J. Bass, K. A. Kloesel, E. C. Gruntfest, and A. Gessner, "Emergency manager decision-making and tornado warning communication," *Meteorological Appl*., vol. 17, no. 2, pp. 163–172, 2010.
[39] WebEOC. [Online]. Available: Manufactured by ESi Acquisition, Inc. http://www.esi911.com/home
[40] E-Teams, by NC4. [Online]. Available: http://www.nc4.us/ETeam.php
[41] National Emergency Management Network. [Online]. Available: http://www.nemn.net/
[42] L. Li and T. Li, "An empirical study of ontology-based multi-document summarization in disaster management," *IEEE Trans. SMC: Syst*., 2013, in press.
[43] L. Tang, T. Li, Y. Jiang, and Z. Chen, "Dynamic query forms for database queries," *IEEE Trans. Knowl. Data Eng*., 2013, in press.



Li Zheng received the B.S. and M.S. degrees in computer science from Sichuan University, Chengdu, China, in 2004 and 2007, respectively. He is currently working toward the Ph.D. degree with the School of Computer Science, Florida International University, Miami, FL, USA.

His research interest includes vertical search engine, recommender system, and disaster management.



Chao Shen received the B.S. and M.S. degrees in computer science from Fudan University, Shanghai, China, in 2006 and 2009, respectively. He is currently working toward the Ph.D. degree with the School of Computer Science, Florida International University, Miami, FL, USA.

His research interest includes text mining, text summarization, and data mining of social media.

Liang Tang received the B.S. and M.S. degrees in computer science from Sichuan University in 2006 and 2009, respectively. He is currently working toward the Ph.D. degree with the School of Computer Science, Florida International University, Miami, FL, USA.

His research interest includes event mining, large scale data mining, and recommender systems.

Chunqiu Zeng received the B.S. and M.S. degrees in computer science from Sichuan University, Chengdu, China, in 2006 and 2009, respectively. He is currently working toward the Ph.D. degree with the School of Computer Science, Florida International University, Miami, FL, USA.

His research interest includes large scale data mining, event mining and text mining.

Tao Li received the Ph.D. degree in computer science in 2004 from the University of Rochester, Rochester, NY, USA.

He is currently an Associate Professor with the School of Computer Science, Florida International University, Miami, FL, USA. His research interests include data mining, machine learning and information retrieval.

Dr. Li received the USA NSF CAREER Award and multiple IBM Faculty Research Awards.

Steve Luis received the Master degree in computer science from Florida International University (FIU), Miami, FL, USA, in 1998.

He is currently the Technical Lead with FIU's Disaster Information Technologies Research Group responsible for the software architecture, requirements, and design for many of the group's core technology tools such as the Business Continuity Information System and the Mobile Disaster Situation Browser. He also conducts business development and partner outreach with more than 100 company and government agencies as part of several public/private partnerships for business recovery in South Florida.

Mr. Luis is recognized for his contribution to the resilience of South Florida communities by the Miami-Dade Emergency Management and Palm Beach County Division of Emergency Management.

Shu-Ching Chen (M'95–SM'04) received the Ph.D. degree in electrical and computer engineering in 1998, and the Master's degrees in computer science, electrical engineering, and civil engineering in 1992, 1995, and 1996, respectively, all from Purdue University, West Lafayette, IN, USA.

He is currently a Full Professor with the School of Computing and Information Sciences, Florida International University, Miami, FL, USA. His main research interests include content-based image/video retrieval, distributed multimedia database management systems, multimedia data mining, multimedia systems, and Disaster Information Management.

Dr. Chen received the 2011 ACM Distinguished Scientist Award. He received the Best Paper Award from the 2006 IEEE International Symposium on Multimedia. He is a Fellow of SIRI and was a steering Committee Member for the IEEE TRANSACTIONS ON MULTIMEDIA from 2011 to 2013.

Multimedia Big Mobile Data Analytics for Emergency Management

Yimin Yang and Shu-Ching Chen

School of Computing and Information Sciences, Florida International University, USA
{yyang010, chens}@cs.fiu.edu

1. Introduction

The world has stepped into a big data era with the development of advanced technologies and the growth of the Internet of Things (IoT). The volume of big data is expected to reach yottabyte (10^{24}) in the near future, among which over 60% will come from wireless mobile devices as opposed to desktops by the year of 2016 [1]. Except for publicly available data released by organizations and government, more and more private individuals begin to share multimedia data through mobile devices across the world. For example, people may take pictures and videos instantly at a disaster scene and share them via social media tools on mobile devices. Accompanying the exponentially growing big data is the challenge of how to analyze and make sense of those data to provide better services to the world. A concrete example is the problem of associating a situation report with plain textual information with the multimedia data collected at a disaster scene to support the timely and efficient decision-making process [2].

With the ever increasing enormous big data, a single-pass framework is infeasible to process the data in real-time. Therefore, many researchers in both industries and academia look for solutions on large-scale data processing. MapReduce (MR) [3] is the framework of choice for large-scale distributed applications. Recent research work in the literature has shown the effectiveness of MR-based frameworks on the tasks of semantic classification [4], information retrieval [5], and so on. More recently, Hadoop Spark [6], a successor system that is more powerful and flexible than Hadoop MapReduce, is merging due to its advantages of lower latency, iterative computation and real-time processing.

In this paper, we propose a multimedia big mobile data computing framework for emergency management. The framework can be described in three stages: the first stage is data collection through vertical search engine and web services; the second stage is data processing, including textual document analysis and multimedia classification leveraging our previous work on the MapReduce Multiple Correspondence Analysis (MR-MCA)-based semantic classification framework; and the third stage is data representation through an iPad application with an intuitive and friendly interface. Section 2 discusses the proposed framework in details. Section 3 presents the system evaluation results and section 4 concludes the paper.

2. A Multimedia Big Mobile Data Framework

We present the proposed multimedia big mobile data framework for emergency management. There are several major components, including data collection, document analysis, key frame and feature extraction, MR-MCA-based classification, report-multimedia association and the final presentation through well-defined user interface as shown in Figure 1.

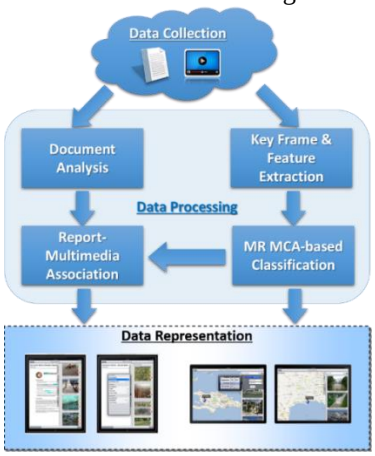


Figure 1. Multimedia big mobile data computing framework for emergency management.

Data Collection.
High-quality, real-time and relevant information is critical for effectively dealing with emergency situations. There are two ways for data collection, i.e., vertical search engine and web services. Specifically, we design and implement a vertical search engine that provides an initial solution for continuously crawling, organizing, indexing and retrieving disaster information. The built in crawler of the vertical search engine runs on a MR cluster. In addition to the data automatically crawled through the vertical search engine, we also provide web services to enable users to upload disaster related data, such as reports, images and videos.

Key Frame and Feature Extraction.
Key frame selection is a critical step for video processing before feature extraction. In this paper, we propose an effective key frame extraction method

based on camera take detection, which can dramatically reduce redundant data in videos while keeping the semantic information.

1)Key frame extraction based on camera take detection
A camera take is a series of consecutive frames taken by a camera. It can be cut into a sequence of segments and interleaved with other camera takes to form a scene which completes an event or a story in a video program. This is a common process in film editing. Figure 2 shows an example of camera take editing results for a video downloaded from FEMA website. Each sub-image in the figure is a key frame selected from a shot (as illustrated in Figure 2(a)), and thus frames (a) to (h) represent consecutive shots, composing a scene. To take a closer look at the key frames, it is obvious that frames (b), (e), and (g) are from the same camera take, so are frames (a) and (c), as well as (f) and (h). Apparently, the shots from the same camera take could be grouped together and represented by one or more frames. It will highly reduce the throughput for further processing.

Figure 2. Examples of camera takes.

Figure 3 depicts the process of camera take detection. Specifically, it takes the following four steps for camera take detection:

- **Frame difference calculation:** based on the assumption that two consecutive frames in a video shot should have a high similarity in terms of visual content. The frame difference is calculated using color histogram (or raw pixel values for saving the computational cost) as a measurement of similarity between two frames.
- **Shot detection:** if the frame difference is above some preset threshold, then a new shot is claimed. The selection of threshold is critical since it may cause over segmentation or down segmentation depending on the types of video programs (action, drama, etc.). To determine a proper threshold and further refine the detection results, certain constraints may apply, such as the shot duration.
- **Key frame selection:** a key frame should properly represent the visual content of a shot. Without loss of generality, the last frame of a shot is selected as the key frame for later processing. It is worth mentioning that more advanced techniques may be utilized to select (or generate) the most representative key frame(s).
- **Camera take detection:** each detected shot (represented by a key frame) will be matched with the last shot in each detected camera take. If a certain matching criterion is satisfied, then the current shot will be added to the end of the matched camera take. It is based on the assumption that a shot is the most related to the one with the closest temporal relationship. Initially, within a certain time period, we may assume the first shot as a camera take. The matching strategies vary from sift point matching to frame difference matching, depending on various performance requirements.

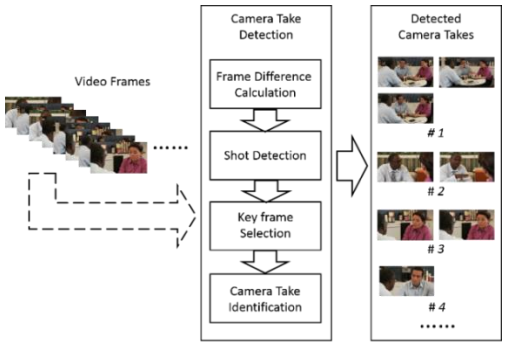


Figure 3. Camera take detection.

2)Feature extraction
Both visual and textual features are extracted for multimedia classification. Specifically, the visual features include visual descriptors, such as Histogram of Oriented Gradient (HOG) [7], color and edge directivity descriptor (CEDD) [8], as well as other low-level visual features, e.g., color histogram, color moment, and texture wavelet [9]. The textual features are extracted from the meta data such as titles and descriptions based on *tf-idf* schema. Except for the aforementioned keyframe-based features, we also extract shot-based features for videos [10]. Since there are totally over 700 dimensional features, some of the features might be redundant or even irrelevant. Therefore, a feature selection step is carried out based on the Hidden Coherent Feature Groups (HCFG) analysis method [9] to identify exemplar features for final classification.

Document Analysis.
Location and subject (e.g., hurricane, flood, and earthquake) are two critical characteristics to describe an emergency event, which are the same for the associated multimedia (such as the images and videos taken at the event scene). The purpose of document analysis is to identify the potential location-subject pairs in a document (e.g., situation report) and relate it to the classified multimedia data. There are mainly three steps for document analysis as follows.

1)Entity extraction
GATE system [11] is a popular tool for natural language processing (NLP). It is applied in our framework to extract tokens and identify certain types of entities, such as data and location, for further analysis.

2)Synonym extraction
Considering the same subject may be expressed in different words in various documents, it is necessary to search all possible synonyms for a specific subject in a document. The synonym extraction is performed using an open source package based on WordNet [12].

3)Location-subject pair identification
After retrieving a list of candidate locations through the GATE system and all synonyms for the corresponding subjects via WordNet, a matching procedure is carried out to identify the final location-subject pairs by examining all the tokens in a document. For more details of the analysis, please refer to [2].

MR-MCA-based Classification.
When dealing with large-scale big mobile data, it is not suitable to use a single-pass framework since the huge volume of data will easily use up memory, not to mention the speed of processing. To accommodate big data requirement and tackle those aforementioned issues, a distributed MCA framework [4] based on MR [3] technique is proposed to perform large-scale correlation-based semantic classification tasks. The MCA algorithm has been proved to be effective for disaster image classification [13], and it was further improved by incorporating temporal information and principle components analysis [10]. In this study, we update MR-MCA and adapt it to video classification as described in Algorithm 1.

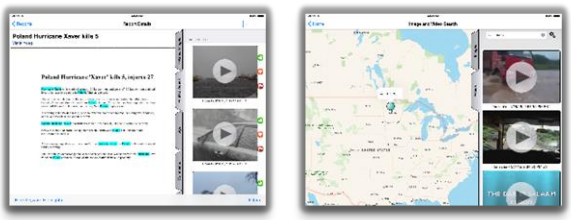
The algorithm receives as its inputs the training and testing data sets, Tr and Te. First the MR-MCA algorithm [4] is applied to Tr to obtain the training model, denoted as $\{F_j, W_j\}_{j=1}^{J}$, where F_j and W_j represent the feature and the corresponding weight set, with J as the total number of features (line 2). Then for each video instance V_i in Te, and each key frame X_q in V_i, the algorithm iterates through the instance's feature items (line 5) and the class labels (line 6, where $|C|$ is the total number of classes), to accumulate the score S_l for each X_q (line 7). The label of X_q is determined by the highest score of S_l, represented as C_l. Finally, the video instance V_i is classified to the one with the largest number of key frames (lines 13-14).

Report-Multimedia Association.
After document analysis and MR-MCA-based classification, we are able to associate the processed situation report with the classified multimedia data based on the identified location-subject pairs from both sides. In the next section, we will introduce the developed iPad application based on the proposed framework, where the functionalities such as filtering based on locations and subjects as well as keywords are provided. The users are also allowed to give feedback to the processed results and to further improve the association.

Algorithm 1 MR-MCA for video classification
Input: Training data set Tr, testing data set Te
Output: Classification results for Te
1: **procedure** CLASSIFICATION(Tr, Te)
2: $\{F_j, W_j\}_{j=1}^{J} \leftarrow$ MR−MCA(Tr);
3: **for** all $V_i \in Te$ $(i = 1, \cdots, N)$ **do**
4: **for** all X_q $(q = 1, \cdots, Q)$ **do**
5: **for** all $F_{j,k} \in \{\{F_k\}^j\}_{j=1}^{J}$ **do**
6: **for** $l \leftarrow 1, \ldots, |C|$ **do**
7: $S_l \leftarrow S_l + abs(w_{j,k}^l)$;
8: **end for**
9: **end for**
10: $C_l \leftarrow \arg\max_l(\{S_l\})$;
11: $\{X_q\}^{C_l} \leftarrow X_q$;
12: **end for**
13: $C_l^* \leftarrow \arg\max_{C_l}(|\{X_q\}^{C_l}|)$;
14: $\{V_i\}^{C_l^*} \leftarrow V_i$;
15: **end for**
16: **return** $\{\{V_i\}^{C_l^*}\}$
17: **end procedure**

3. System Evaluation

Since the evaluation of report-image association has been conducted in our previous work [2], we will mainly evaluate the report-video association in this paper. We have crawled over 1500 videos from the Internet and processed the data using the proposed framework. Figure 4 shows two of the major interfaces in our developed iPad application, where Figure 4(a) shows the situation report with related videos that are classified based on the pre-defined ontology. The users are able to filter the videos based on locations, subjects, and keywords. In addition, users can offer feedback to the retrieved results and the system will automatically refine the association accordingly. Furthermore, we also provide the functionality of retrieving disaster videos using keywords and displaying them on a map based on locations as shown in Figure 4(b).



(a) (b)
Figure 4. iPad application interfaces.

4. Conclusion

In this paper, we have presented a multimedia big mobile data computing framework consisting of three stages, namely data collection, data processing, and data representation. In the data collection stage, vertical search engine and web services are used to collect high quality disaster data. In the data processing stage, multimedia data analytics implemented on top of the MR framework are performed to associate situation reports (in plain texts) with multimedia data (such as images and videos). Finally, an iPad application with interactive interfaces is developed.

Acknowledgement

This research is partially supported by DHS under grant Award Number 2010-ST-062-000039, DHS's VACCINE Center under Award Number 2009-ST-061-CI0001, NSF HRD-0833093 and CNS-1126619. The authors would like to thank Xiaoyu Dong for his assistance in collecting data and implementing the iPad application.

References

[1] http://venturebeat.com/2015/01/22/big-data-and-mobile-analytics-ready-to-rule-2015/.

[2] Y. Yang, W. Lu, J. Domack, T. Li, S.-C. Chen, S. Luis, and J. K. Navlakha, "MADIS: A Multimedia-Aided Disaster information Integration System for emergency management," *The 8th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom)*, pp. 233-241. 2012.

[3] J. Dean and G. Sanjay, "MapReduce: Simplified Data Processing on Large Clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107-113, 2008.

[4] F. C. Fleites, H.-Y. Ha, Y. Yang, and S.-C. Chen, "Large-Scale Correlation-Based Semantic Classification Using MapReduce," *Cloud Computing and Digital Media: Fundamentals, Techniques, and Applications*, pp. 169-190, CRC Press, 2014.

[5] S. P. Dravyakar, S. B. Mane, and P. K. Sinha, "Private Content Based Multimedia Information Retrieval Using Map-Reduce," *International Journal of Computer Science Engineering and Technology (IJCSET)*, vol. 4, no. 4, pp. 125-128, 2014.

[6] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Cluster Computing with Working Sets," *Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing*, pp. 10-10, 2010.

[7] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 886-893, 2005.

[8] S. A. Chatzichristofis and Y. S. Boutalis, "CEDD: Color and Edge Directivity Descriptor: A Compact Descriptor for Image Indexing and Retrieval," *Computer Vision Systems*, pp. 312-322, 2008.

[9] Y. Yang, H.-Y. Ha, F. C. Fleites, and S.-C. Chen, "A Multimedia Semantic Retrieval Mobile System Based on HCFGs," in *IEEE MultiMedia*, vol. 21, no. 1, pp. 36-46, 2014.

[10] Y. Yang, S.-C. Chen, and M.-L. Shyu, "Temporal Multiple Correspondence Analysis for Big Data Mining in Soccer Videos," *The First IEEE International Conference on Multimedia Big Data (BigMM)*, 2015.

[11] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan, "Gate: A Framework and Graphical Development Environment for Robust NLP Tools and Applications," *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 168-175, 2002.

[12] Princeton University. Wordnet, A Lexical Database for English. http://wordnet.princeton.edu/, July 2011.

[13] Y. Yang, H.-Y. Ha, F. C. Fleites, S.-C. Chen, and S. Luis, "Hierarchical Disaster Image Classification for Situation Report Enhancement," *The 12th IEEE International Conference on Information Reuse and Integration (IRI)*, pp. 181-186, 2011.

Yimin Yang is a Ph.D. candidate at the School of Computing and Information Sciences (SCIS), Florida International University (FIU), Miami. She received her M.S. degree in Computer Science from FIU in 2012. Her research interests include multimedia data mining, multimedia systems, image and video processing, and information retrieval.

Shu-Ching Chen is an Eminent Scholar Chaired Professor in Computer Science at the School of Computing and Information Sciences, Florida International University, Miami since August 2009. Prior to that, he was an Assistant/Associate Professor in SCIS at FIU from 1999. He received his Ph.D. degree in Electrical and Computer Engineering in 1998, and Master's degrees in Computer Science, Electrical Engineering, and Civil Engineering in 1992, 1995, and 1996, respectively, all from Purdue University, West Lafayette, IN, USA. His main research interests include content-based image/video retrieval, multimedia data mining, multimedia systems, and Disaster Information Management. Dr. Chen was named a 2011 recipient of the ACM Distinguished Scientist Award. He received the best paper award from 2006 IEEE International Symposium on Multimedia. He was awarded the IEEE Systems, Man, and Cybernetics (SMC) Society's Outstanding Contribution Award in 2005 and was the co-recipient of the IEEE Most Active SMC Technical Committee Award in 2006.

Ensemble Contextual Bandits for Personalized Recommendation

Liang Tang Yexi Jiang Lei Li Tao Li
School of Computing and Information Sciences
Florida International University
11200 S.W. 8th Street, Miami, FL 33199
{ltang002,yjian004,lli003,taoli}@cs.fiu.edu

ABSTRACT

The cold-start problem has attracted extensive attention among various online services that provide personalized recommendation. Many online vendors employ contextual bandit strategies to tackle the so-called *exploration/exploitation* dilemma rooted from the cold-start problem. However, due to high-dimensional user/item features and the underlying characteristics of bandit policies, it is often difficult for service providers to obtain and deploy an appropriate algorithm to achieve acceptable and robust economic profit.

In this paper, we explore ensemble strategies of contextual bandit algorithms to obtain robust predicted click-through rate (CTR) of web objects. The ensemble is acquired by aggregating different pulling policies of bandit algorithms, rather than forcing the agreement of prediction results or learning a unified predictive model. To this end, we employ a meta-bandit paradigm that places a hyper bandit over the base bandits, to explicitly explore/exploit the relative importance of base bandits based on user feedbacks. Extensive empirical experiments on two real-world data sets (news recommendation and online advertising) demonstrate the effectiveness of our proposed approach in terms of CTR.

Categories and Subject Descriptors:
H.3.3 [Information Search and Retrieval]: Information Filtering; H.3.5 [Information Systems]: On-line Information Services; I.2.6 [Computing Methodologies]: Learning

Keywords: Personalized Recommendation; Ensemble Recommendation; Contextual Bandit; CTR Prediction; Meta Learning

1. INTRODUCTION

Personalized recommendation services aim to identify popular items and tailor the content according to users' preferences. In practice, a large number of users or items might be completely new to the system, which is referred to as the *cold-start* problem [21]. This issue is often recognized as an exploration/exploitation problem, in which we have to find a trade-off between two competing goals: maximizing users' satisfaction in a long run, while exploring uncertainties of user interests [1]. For example, a news recommender should provide breaking news to users while maintaining user preferences based on aging news stories.

The aforementioned issue is often modeled as a contextual bandit problem [29]. A contextual bandit problem is a series of trials with a fixed number of arms. In each trial, the algorithm selects an arm to pull based on the given context. By pulling an arm, it can obtain a reward, drawn from some unknown distribution determined by the pulled arm and the context. The objective of bandit algorithms is to maximize the total obtained reward. In the cold-start situation, a recommender system does not have enough training data to build the predictive model. In such a case, people often use a bandit algorithm to solve the recommendation problem, where each trial can be treated as a user visit, and each arm is an item (e.g., a news article or advertisement). Pulling an arm is recommending that item, where the context is a set of user features. The reward is the user response (e.g., a click), which is also determined by the recommended item and user features. The objective of recommender systems is to maximize the total user response, which is equivalent to maximizing the total reward in bandit algorithms [13].

Recently, a series of algorithms have been reported to tackle the multi-armed bandit problem, including unguided exploration (e.g., ϵ-greedy [28], and epoch-greedy [12]), guided exploration (e.g., LinUCB [13], EXP4 [2], and Thompson sampling [6]). These existing algorithms can achieve promising performance under specific settings. The performances of different policies vary significantly in many recommendation applications. A common practice of picking the appropriate policy is to first evaluate these policies and then select the best one to deploy. However, in the cold-start situation, it is often difficult to conduct an unbiased offline evaluation due to the deficiency of the historical data. For online evaluation, e.g., A/B test, the user visit traffic has to be split to multiple buckets for different policies, and therefore the number of testing policies running in parallel is restricted in order to obtain acceptable daily income.

In our work, we explore the possibility of utilizing ensemble strategies to obtain a robust policy that can achieve acceptable CTR in various recommender systems. As the predictive result of each contextual bandit algorithm is the pulled arm (item), it is not appropriate to adopt the majority voting or consensus prediction as the ensemble. We hence resort to meta learning to build a hyper policy that adaptively allocates the pulling chances to different base policies

RecSys'14, October 6–10, 2014, Foster City, Silicon Valley, CA, USA.

http://dx.doi.org/10.1145/2645710.2645732.

based on the estimation of their performance. The proposed ensemble bandit algorithms may not produce the optimal CTR of base policies, but it can always approach to the best one, which gives a robust mechanism for online personalized recommendation in the cold-start situations. In summary, the contribution of our work is three-fold:

- We explore the possibility of stabilizing the CTR estimation of recommender systems by integrating the advantages of different bandit policies.
- We propose two ensemble strategies to address the cold-start problem in personalized recommender systems, which employ a meta-bandit learning paradigm to achieve the robustness of the CTR.
- We conduct extensive experiments on real-world data sets to demonstrate the efficacy of the proposed method compared with other baseline algorithms. The results show that our method is robust in terms of CTR.

The rest of this paper is organized as follows. In Section 2, we describe a brief summary of prior work relevant to contextual bandit problems, ensemble recommendation and meta learning. We then formulate the problem in Section 3, and present the detailed algorithmic description in Section 4. Extensive empirical evaluation results are reported in Section 5. Finally, Section 6 concludes the paper.

2. RELATED WORK

In our work, we employ ensemble strategies combined with a meta learning paradigm to stabilize the output of contextual bandit algorithms. In the following, we highlight the previous research that are most relevant to our work.

Contextual Bandits: When predicting the CTR of web data, the *cold-start* problem is often modeled as a contextual bandit problem with exploration/exploitation trade-off, where user features are regarded as contextual information. Typical solutions of this problem involve unguided exploration (e.g., ϵ-greedy [28], epoch-greedy [12]), guided exploration (e.g., LinUCB [13], EXP4 [2]) and probability matching (e.g., Thompson sampling [6, 22]). Most existing methods require either a parameter to control the importance of exploration or prior information of Bayesian learning models; however in practice, it is difficult to determine the optimal value for the input due to the insufficiency of user feedbacks. Hence, the prediction performance of these algorithms is not stable along both exploration and exploitation phases, unless the selection policy/model converges.

Ensemble Recommendation: Ensemble based algorithms have been well explored to improve the performance of prediction [20, 27], and are often preferred in recommendation competitions, such as the Netflix Prize contest [11, 24] and KDD Cups [17, 30]. Typically, an ensemble method combines the prediction of different algorithms to obtain a final prediction [18], which is often referred to as "blending" [10]. The most basic blending strategy is to acquire the final prediction based on the mean over all the prediction results or the majority vote. Learning based approaches have also been proposed to unify different recommendation algorithms [31]. In our work, to obtain a robust policy, we resort to ensemble strategies that assimilate the advantages of different contextual bandit algorithms.

Meta Learning: In machine learning community, the goal of meta learning is to accumulate experience on the performance of multiple learning algorithms [8]. Meta learning has been widely used in algorithm selection [16, 19]. Due to the uncertainty of learning algorithms, i.e., we do not know in advance the predictive performance, a lot of work has modeled algorithm selection as a multi-armed bandit problem [7, 25] and tries to balance the trade-off between exploring algorithm capabilities and exploiting the predictive power of algorithms. In addition, some recent research efforts [15, 23] focus on meta learning of exploration/exploitation strategies, where the base learners are bandit algorithms.

3. PROBLEM FORMULATION

In this section, we formulate the problem studied in this paper. Let $\mathcal{A}$ denote the set of items (or bandit arms), $\mathcal{A} = \{a_1, ..., a_k\}$, and $\Pi = \{\pi_1,...,\pi_m\}$ be a given set of recommenders (or policies), where each recommender is a contextual bandit algorithm with a specific parameter setting. $\pi_i(\mathbf{x}) = a$ indicates that the policy π_i pulls a with respect to $\mathbf{x}$, where $a \in \mathcal{A}$ and $\mathbf{x}$ is a context feature vector. Let $\mathcal{D}$ be the space of $\mathbf{x}$ and $p(\mathbf{x})$ denote the probability density of $\mathbf{x}$. After the pulling, π_i receives a reward r, where r is a value drawn from the conditional distribution $p(\cdot|\mathbf{x}, a)$. In our work, we only consider $r \in \{0, 1\}$, i.e., a non-click/click on a specific item. Thus, $p(\cdot|\mathbf{x}, a)$ is a Bernoulli distribution.

Each policy $\pi_i \in \Pi$ aims to maximize the expected received reward denoted by $\mathrm{E}[r_{\pi_i}]$, where

$$\begin{aligned}\mathrm{E}[r_{\pi_i}] &= \int_{\mathcal{D}} \sum_{a\in\mathcal{A}} \sum_{r\in\{0,1\}} r \cdot p(r|\mathbf{x}, a) \cdot p(a = \pi_i(\mathbf{x})|\mathbf{x}) \cdot p(\mathbf{x})\, \mathrm{d}\mathbf{x} \\ &= \int_{\mathcal{D}} \sum_{\pi_i(\mathbf{x})\in\mathcal{A}} \sum_{r\in\{0,1\}} r \cdot p(r, \pi_i(\mathbf{x}), \mathbf{x})\, \mathrm{d}\mathbf{x}.\end{aligned}$$

The expected reward $\mathrm{E}[r_{\pi_i}]$ is known as the CTR of the policy π_i, which is often used as the performance metric.

Existing studies propose different contextual bandit algorithms and show their empirical performances on various real-world data sets [5, 6, 13]. It is known that a bandit algorithm with different parameter settings can have different performance [22]. The choice of parameters depends on the distribution of the real data, which is often unknown in the *cold-start* situation. Therefore, given a set of policies Π, individual polices in Π can have very different performance for a particular recommender system. Let π^* denote the best policy in terms of the performance, i.e.,

$$\pi^* = \arg\max_{\pi_i\in\Pi} \mathrm{E}[r_{\pi_i}].$$

For different recommendation problems, π^* is different and not known in advance. The goal of this paper is to develop an ensemble contextual bandit policy such that its performance can be close to the performance of π^*.

4. ALGORITHM

This section presents two ensemble bandit algorithms, `HyperTS` and `HyperTSFB`, for solving the contextual recommendation problem in the cold-start situation. The idea of these two algorithms is to distribute the trials to the base bandit policies. Given a set of policies $\Pi = \{\pi_1, ..., \pi_m\}$ and a context $\mathbf{x}$, both algorithms make two decisions to deicide which arm to pull:

- Decision 1. Select a policy from Π, denoted by π_i;
- Decision 2. Select the arm $a = \pi_i(\mathbf{x})$ to pull.

For Decision 1, if we know which base policy is the best one, i.e., π^*, we can always select it. However, the performance of each policy is unknown at the beginning. To estimate their expected rewards, we need to select them and observe the received rewards. Same as Decision 2, the exploration-exploitation dilemma also exists in Decision 1. In this paper, we focus on Decision 1.

To address the policy selection problem in Decision 1, both of the proposed algorithms leverage non-contextual Thompson sampling [6, 26]. Generally, in each trial, the algorithms randomly select a policy $\pi_i \in \Pi$, where the probability of selecting π_i is equal to the probability of π_i being π^*, i.e., $p(\hat{\mathrm{E}}[r_{\pi_i}] = \max_{\pi_j \in \Pi} \hat{\mathrm{E}}[r_{\pi_j}])$, where $\hat{\mathrm{E}}[r_{\pi_i}]$ is the estimated expected reward of the policy π_i. It is difficult to directly compute this probability [22]. Thus, in each trial, we randomly draw a value, denoted by $\tilde{r}_{\pi_i}$, from the distribution of $\hat{\mathrm{E}}[r_{\pi_i}]$ for each $\pi_i \in \Pi$, and then select the policy that has the maximum value of $\tilde{r}_{\pi_i}$. In the following, we present two approaches to estimate $\mathrm{E}[r_{\pi_i}]$.

4.1 HyperTS Algorithm

HyperTS estimates the expected reward $\mathrm{E}[r_{\pi_i}]$ of each policy $\pi_i \in \Pi$ using Monte Carlo method. Concretely, let $\mathbf{x}_1$,...,$\mathbf{x}_n$ be the contexts of n trials in which π_i is selected. $\mathbf{x}_1$,...,$\mathbf{x}_n$ are samples drawn from $p(\mathbf{x})$. For an input context $\mathbf{x}_j$, π_i pulls the arm a_j and receives the reward r_j, where a_j is seen as a sample from $p(a = \pi_i(\mathbf{x}_j)|\mathbf{x}_j)$, r_j is seen as a sample drawn from $p(r|\mathbf{x}_j, a_j)$. Thus, $(\mathbf{x}_j, a_j, r_j)$ is a sample drawn from the joint distribution $p(\mathbf{x}, a, r)$, $j = 1, ..., n$. The Monte Carlo estimate is $\hat{\mathrm{E}}[r_{\pi_i}] = \frac{1}{n}\sum_{j=1}^{n} r_j$. The rewards $r_1, ..., r_n \in \{0, 1\}$ are the sample drawn from the Bernoulli distribution $p(r)$, which is a marginal distribution of $p(\mathbf{x}, a, r)$. Therefore, $\hat{\mathrm{E}}[r_{\pi_i}]$ follows the Beta distribution:

$$\hat{\mathrm{E}}[r_{\pi_i}] \sim \mathrm{Beta}(1 + \alpha_{\pi_i}, 1 + \beta_{\pi_i}),$$

where $\alpha_{\pi_i} = \sum_{j=1}^{n} r_j$ and $\beta_{\pi_i} = n - \alpha_{\pi_i}$. For the prior, $\mathrm{Beta}(1, 1)$ is used. Algorithm 1 shows the pseudo-code of `HyperTS`. In each trial, r_i is a sample of $\hat{\mathrm{E}}[r_{\pi_i}]$, drawn from a Beta distribution. The selected policy is the one having the maximum r_i.

Algorithm 1 `HyperTS`(Π)

Input: Π : the set of base bandit policies.
for $i = 1, ..., m$ **do**
  $\alpha_{\pi_i} \leftarrow 0$, $\beta_{\pi_i} \leftarrow 0$
end for
for $t = 1, 2, ...$ **do**
  for $i = 1, ..., m$ **do**
    Draw r_i from $\mathrm{Beta}(1 + \alpha_{\pi_i}, 1 + \beta_{\pi_i})$.
  end for
  Pull the arm $a = \pi_j(\mathbf{x}_t)$, where $j = \arg\max_{i=1,...,m} r_i$.
  Receive the reward $r_{\mathbf{x}_t,a}$ and feed π_j with $r_{\mathbf{x}_t,a}$.
  if $r_{\mathbf{x}_t,a} = 1$ **then**
    $\alpha_{\pi_j} \leftarrow \alpha_{\pi_j} + 1$
  else
    $\beta_{\pi_j} \leftarrow \beta_{\pi_j} + 1$
  end if
end for

4.2 HyperTSFB Algorithm

In `HyperTS`, the expected reward of each base policy is estimated only from the feedback when that policy is selected. The feedback of the decision made by other policies is not utilized. If the number of policies in Π is large, the total number of trials needed for exploring the performance of base policies will be large and the total reward will be smaller. To improve the estimation efficiency, we propose `HyperTSFB` (`HyperTS` with shared feedback), an algorithm that fully utilizes every received feedback for expected reward estimation.

Given the context $\mathbf{x}$, `HyperTSFB` requires each base policy $\pi_i \in \Pi$ provide the probability of π_i pulling the arm a, i.e., $p(a = \pi_i(\mathbf{x})|\mathbf{x})$. Then, even though the policy π_i is not selected in the trial, `HyperTSFB` can still utilizes the feedback for $\mathbf{x}$ to estimate the expected reward of π_i. For some policy, $p(a = \pi_i(\mathbf{x})|\mathbf{x})$ can be computed directly. For instance, if π_i denotes `random` policy, then $p(a = \pi_i(\mathbf{x})|\mathbf{x}) = 1/k$, $k = |\mathcal{A}|$; if π_i is ϵ-`greedy`, then

$$p(a = \pi_i(\mathbf{x})|\mathbf{x}) = \begin{cases} \epsilon/k + (1 - \epsilon) & \text{if } a = a^* \\ \epsilon/k & \text{if } a \neq a^*, \end{cases}$$

where a^* is the arm that has the maximum predicted reward by the input $\mathbf{x}$. For some policy, $p(a = \pi_i(\mathbf{x})|\mathbf{x})$ can be difficult to compute, e.g. contextual Thompson sampling. We can invoke $\pi_i(\mathbf{x})$ multiple times to estimate this probability according to the frequency of a being output.

Once we compute $p(a = \pi_i(\mathbf{x})|\mathbf{x})$, $\mathrm{E}[r_{\pi_i}]$ can be rewritten as Eq.(1) and importance sampling [32] can be leveraged for estimation.

$$\begin{aligned}
\mathrm{E}[r_{\pi_i}] &= \sum_{a \in \mathcal{A}} \int_{\mathcal{D}} \sum_{r \in \{0,1\}} r \cdot p(r|\mathbf{x}, a) p(a = \pi_i(\mathbf{x})|\mathbf{x}) p(\mathbf{x}) \, \mathrm{d}\mathbf{x} \\
&= \sum_{a \in \mathcal{A}} \int_{\mathcal{D}} \sum_{r \in \{0,1\}} r \frac{p(a = \pi_i(\mathbf{x})|\mathbf{x})}{p(a|\mathbf{x})} p(r|\mathbf{x}, a) p(a|\mathbf{x}) p(\mathbf{x}) \, \mathrm{d}\mathbf{x} \\
&= \sum_{a \in \mathcal{A}} \int_{\mathcal{D}} \sum_{r \in \{0,1\}} (r \cdot w_{a,\mathbf{x}}^{\pi_i}) \cdot p(r|\mathbf{x}, a) p(a|\mathbf{x}) p(\mathbf{x}) \, \mathrm{d}\mathbf{x} \\
&= \sum_{a \in \mathcal{A}} p(a) \int_{\mathcal{D}} \sum_{r \in \{0,1\}} (r \cdot w_{a,\mathbf{x}}^{\pi_i}) \cdot p(r, \mathbf{x}|a) \, \mathrm{d}\mathbf{x} \\
&= \sum_{a \in \mathcal{A}} p(a) \, \mathrm{E}_{r,\mathbf{x}}[r \cdot w_{a,\mathbf{x}}^{\pi_i}|a]. \qquad (1)
\end{aligned}$$

In Eq.(1), $w_{a,\mathbf{x}}^{\pi_i} = \frac{p(a=\pi_i(\mathbf{x})|\mathbf{x})}{p(a|\mathbf{x})}$ is the importance weight, and $\mathrm{E}_{r,\mathbf{x}}[r \cdot w_{a,\mathbf{x}}^{\pi_i}|a]$ is the expected reward of π_i by pulling a. Eq.(1) states that the expected reward estimation can be separated into multiple estimations for different arms. In the importance weight, $p(a|\mathbf{x})$ is the probability of the arm a being pulled given the context $\mathbf{x}$,

$$p(a|\mathbf{x}) = \sum_{\pi_i \in \Pi} p(a = \pi_i(\mathbf{x})|\mathbf{x}) p(\pi_i = \arg\max_{\pi_j \in \Pi} \hat{\mathrm{E}}[r_{\pi_j}]).$$

In the implementation, we use a sampling-based method to obtain the value of $p(a|\mathbf{x})$. For each given context $\mathbf{x}$, we invoke `HyperTSFB` multiple times and then estimate $p(a|x)$ according to the frequency of a being selected. $p(a)$ is the marginal probability of a being selected, which is simply approximated by the ratio of a being pulled in all previous trials done by `HyperTSFB`.

In Eq.(1), the expected reward of π_i by pulling a is

$$\begin{aligned}
\mathrm{E}_{r,\mathbf{x}}[r \cdot w_{a,\mathbf{x}}^{\pi_i}|a] &= \int_{-\infty}^{+\infty} y \cdot p(r \cdot w_{a,\mathbf{x}}^{\pi_i} = y|a)\,\mathrm{d}y \\
&= \int_{-\infty}^{+\infty} y \cdot p(r=1, w_{a,\mathbf{x}}^{\pi_i} = y|a)\,\mathrm{d}y \\
&= \int_{-\infty}^{+\infty} y \cdot p(w_{a,\mathbf{x}}^{\pi_i} = y|r=1,a)p(r=1|a)\,\mathrm{d}y \\
&= p(r=1|a) \cdot \int_{-\infty}^{+\infty} y \cdot p(w_{a,\mathbf{x}}^{\pi_i} = y|r=1,a)\,\mathrm{d}y \\
&= \mathrm{E}_{r,\mathbf{x}}[r|a] \cdot \mathrm{E}_{\mathbf{x}}[w_{a,\mathbf{x}}^{\pi_i}|r=1,a]. \qquad (2)
\end{aligned}$$

In Eq.(2), $\mathrm{E}_{r,\mathbf{x}}[r|a]$ is the expected reward of pulling arm a, which is also the overall CTR of a and determined by the popularity of a. The importance weight $w_{a,\mathbf{x}}^{\pi_i}$ is proportional to the probability of π_i pulling a given $\mathbf{x}$. $\mathrm{E}_{\mathbf{x}}[w_{a,\mathbf{x}}^{\pi_i}|r=1,a]$ reflects how likely will π_i pulls a if the reward of a is 1. Intuitively, Eq.(2) states that the expected reward of π_i by pulling a is determined by the popularity of a and the likelihood of π_i pulling a if the reward of a is 1. Since $r \in \{0,1\}$, we use a Beta distribution to model $\hat{\mathrm{E}}_{r,\mathbf{x}}[r|a]$, i.e.

$$\hat{\mathrm{E}}_{r,\mathbf{x}}[r|a] \sim \mathrm{Beta}(1+\alpha_a, 1+\beta_a),$$

where α_a is the number of trials that the received reward is 1 by pulling a, β_a is the number of trials that the received reward is 0 by pulling a. For the prior, uniform distribution $\mathrm{Beta}(1,1)$ is used. $\mathrm{E}_{\mathbf{x}}[w_{a,\mathbf{x}}^{\pi_i}|r=1,a]$ is estimated by the mean of importance weights in previous trials in which a is pulled and the reward is 1. Assuming for one policy and one arm those importance weights will converge in one distribution, based on Central Limit Theory, the sample mean follows a normal distribution when the sample size is sufficient large, i.e.

$$\hat{\mathrm{E}}_{\mathbf{x}}[w_{a,\mathbf{x}}^{\pi_i}|r=1,a] \sim \mathcal{N}(\mu_{a,i}, \sigma_{a,i}^2/n_{a,i}),$$

where $\mu_{a,i}$ and $\sigma_{a,i}^2$ are the mean and variance of the distribution of the importance weights, $n_{a,i}$ is the sample size to calculate the the mean and variance. If the sample size $n_{a,i}$ is not sufficient large (less than 30 [9]), we draw $\hat{\mathrm{E}}_{\mathbf{x}}[w_{a,\mathbf{x}}^{\pi_i}|r=1,a]$ from uniform distribution $\mathcal{U}(0,1)$.

Algorithm 2 shows the pseudo-code of `HyperTSFB`. For trial $t = 1, 2, ...$, given the context $\mathbf{x}_t$, the sampled expected reward of π_i is r_{π_i}, which is calculated based on two other sampled values from each arm (Line 8 to 21). As for the received reward, $r_{\mathbf{x}_t,a}$, the estimated parameters of every base policy and arm a are updated (Line 24 to 34).

Algorithm 2 `HyperTSFB`(Π)

Input: Π : the set of base bandit policies.

1: **for** $j = 1, ..., k$ **do**
2: $\alpha_{a_j} \leftarrow 0$, $\beta_{a_j} \leftarrow 0$, $n_{a_j} \leftarrow 0$
3: **for** $i = 1, ..., m$ **do**
4: $n_{a_j,i} \leftarrow 0$
5: **end for**
6: **end for**
7: **for** $t = 1, 2, ...$ **do**
8: **for** $j = 1, ..., k$ **do**
9: Draw r_{a_j} from $\mathrm{Beta}(1+\alpha_{a_j}, 1+\beta_{a_j})$
10: **end for**
11: **for** $i = 1, ..., m$ **do**
12: $r_{\pi_i} \leftarrow 0$
13: **for** $j = 1, ..., k$ **do**
14: **if** $n_{a_j,i} < 30$ **then**
15: Draw $w_{a_j}^{\pi_i}$ from $\mathcal{U}(0,1)$
16: **else**
17: Draw $w_{a_j}^{\pi_i}$ from $\mathcal{N}(\mu_{a_j,i}, \sigma_{a_j,i}^2/n_{a_j,i})$
18: **end if**
19: $r_{\pi_i} \leftarrow r_{\pi_i} + n_{a_j}/t \cdot r_{a_j} \cdot w_{a_j}^{\pi_i}$
20: **end for**
21: **end for**
22: Pull the arm $a = \pi_{s^*}(\mathbf{x}_t)$, where $s^* = \arg\max_{i=1,...,m} r_{\pi_i}$.
23: Receive the reward $r_{\mathbf{x}_t,a}$, feed each $\pi_i \in \Pi$ with $r_{\mathbf{x}_t,a}$
24: $n_{a_j} \leftarrow n_{a_j} + 1$
25: **if** $r_{\mathbf{x}_t,a} = 1$ **then**
26: **for** $i = 1, ..., m$ **do**
27: $w_t \leftarrow \frac{p(a=\pi_i(\mathbf{x}_t)|\mathbf{x}_t)}{p(a|\mathbf{x}_t)}$
28: Update $\mu_{a,i}$ and $\sigma_{a,i}^2$ by w_t
29: $n_{a,i} \leftarrow n_{a,i} + 1$
30: **end for**
31: $\alpha_a \leftarrow \alpha_a + 1$
32: **else**
33: $\beta_a \leftarrow \beta_a + 1$
34: **end if**
35: **end for**

5. EVALUATION

5.1 Data Collections

We verify our approaches on two real-world data sets, including news recommendation data (*Yahoo! Today News*) and online advertising data (*KDD Cup* 2012, Track 2).

Yahoo! Today News data set is collected by Yahoo! Today module[1]. News articles were randomly displayed on the *Yahoo! Front Page* from October 2nd, 2011 to October 16th, 2011. The data set contains 28,041,015 user visit events to the *Yahoo! Today Module*. Each visit event is associated with the user's information, e.g., age, gender, behavior targeting features, etc., represented by a binary feature vector of dimension 136. This data set has been used for evaluating contextual bandit algorithms in other literatures [6, 13, 14]. 10 million user visit events are used in this evaluation.

KDD Cup Online Advertising data set is published by *KDD Cup 2012*[2]. Each record of this data set is an ad impression, containing user profile, queries, ad information and click counts. In our work, the context is represented as a binary feature vector, each entry of which denotes whether a query token is contained in the search query. User profiles, e.g., gender and age, are also appended to the context vector using the binary format. The dimension of the context features for this data set is 1,070,866. One issue of this data set is that the click information is extremely sparse due to the large pool of the ads. To alleviate this problem, we only select the top 50 ads with the most impressions. The generated data set contains 9 million user visit events.

5.2 Evaluation Methods

The experiments on *Yahoo! Today News* data set is evaluated by the *Replayer* method [14], which provides an unbiased offline evaluation by utilizing the historical log. It has been shown that the CTR estimated by this *Replayer* approaches the real CTR of the deployed online system if

[1]http://webscope.sandbox.yahoo.com/catalog.php.

[2]http://www.kddcup2012.org/c/kddcup2012-track2.

the items in historical user visits are randomly and uniformly recommended [14]. However, for *KDD Cup* data set, the search ads in historical logs are not uniformly recommended. We hence evaluate this data set using a simulation method [6]. In the simulation method, we first train a logistic regression model for each ad using the entire data offline. Then, for each impression with context $\mathbf{x}$, the click of an ad is generated with a probability $(1+\exp(-\mathbf{w}^T\mathbf{x}))^{-1}$. Although the evaluated CTR is not the real CTR, it provides a methodology for comparing different policies in such high dimensional data.

5.3 Experimental Setup

For evaluation purpose, we use the averaged reward as the metric, which is the total reward divided by the total number of trials, i.e., $\frac{1}{n}\sum_{t=1}^{n} r_t$, where n is the number of trials. The higher the CTR, the better the performance. In the experiments, to avoid the leakage of business-sensitive information, we report the relative CTR, which is the overall CTR of an algorithm divided by the overall CTR of random selection. The base policies used for the ensemble contain multiple types of algorithms, including:

- `Random`: it randomly selects an arm to pull.
- `ϵ-greedy` (ϵ): it randomly selects an arm with probability ϵ and selects the arm of the largest predicted reward with probability $1-\epsilon$.
- `LinUCB` (α) [13]: In each trial, it pulls the arm of the largest score, which is a linear combination of the mean and standard deviation of the predicted reward. Given a context $\mathbf{x}$, the score is $\hat{\boldsymbol{\mu}}^T\mathbf{x} + \alpha\sqrt{\mathbf{x}^T\hat{\boldsymbol{\Sigma}}^{-1}\mathbf{x}}$, where $\hat{\boldsymbol{\mu}}$ and $\hat{\boldsymbol{\Sigma}}$ are the estimated mean and covariance of the posterior distribution, and α is a predefined parameter for controlling the balance of exploration and exploitation.
- `Softmax` (τ) [3]: it randomly selects an arm a_i with probability $\frac{\exp(r_{a_i}/\tau)}{\sum_{a_j\in\mathcal{A}}\exp(r_{a_j}/\tau)}$, where r_{a_i} is the predicted reward for the arm a_i.
- `Epoch-greedy` [12]: it defines an epoch with length L, in which a one-step exploration is first performed, and the rest trials in the epoch are used for exploitation.
- `TS` (q_0) [6]: thompson sampling with logistic regression, which randomly draws the coefficients from the posterior distribution, and selects the arm of the largest predicted reward. The priori distribution is $\mathcal{N}(\mathbf{0}, q_0^{-1}\mathbf{I})$.
- `TSNR` (q_0): it is similar to `TS`(q_0), but in the stochastic gradient ascent, there is no regularization by the prior. The priori distribution $\mathcal{N}(\mathbf{0}, q_0^{-1}\mathbf{I})$ is only used in the calculation of the posterior distribution for the parameter sampling, but not in the learning algorithm.

In the experiments, the reward in a single recommendation activity is the user click, which is a binary value. Therefore, *logistic regression* is applied as the learning model in all policies (except for `Random`). Since the contextual bandit algorithms are online algorithms, *stochastic gradient ascent* is used as the learning algorithm [4]. Notice that the algorithms digest the data in an online manner, hence all the user visits in the data sets are used for the testing purpose.

In our problem setting, we utilize ensemble strategies to obtain a unified policy. To evaluate the effectiveness of the ensemble, we empirically compare the following ensemble algorithms:

- `HyperRandom`: it randomly selects a policy from the policy pool, and then performs the recommendation based on the selected policy.
- `HyperTS`: The one proposed in Section 4.1.
- `HyperTSFB`: The one proposed in Section 4.2.

5.4 Result Analysis

For each policy, we test its performance on the entire data to obtain the overall CTR. To emphasize the robustness of our proposed ensemble strategy, we also split the data into multiple time buckets, and evaluate how the policies perform on each individual time bucket.

5.4.1 On Overall CTR

For base policies, most of them are randomized except for `LinUCB`. For each trial, we randomly shuffle the pool of items to be recommended. Thus, the performance of `LinUCB` may vary in different runs. We run each policy 10 times, and calculate the mean, standard deviation, minimum and maximum of the overall CTR. Table 1 reports the results. The mean values of the top 5 base policies are highlighted in **bold**, and the comparable mean values of ensemble strategies are emphasized by **bold***.

As depicted in the table, the performance of the base policies varies significantly with different parameter values. Except for `Random` (pure exploration) and ϵ-greedy(0.0) (pure exploitation), all the base policies take into account both exploration and exploitation. If the parameter that controls the relative importance of exploration and exploitation is perfectly set, then the performance approaches to the optimal; otherwise, the performance is relatively poor. However in practice, it is often difficult to determine the optimal value for the input parameter of each policy, primarily due to the online learning process of the algorithms as well as the unknown data distribution for learning models.

Our proposed ensemble strategies can achieve comparable performance with the top ranked policies in terms of the overall CTR, by virtue of the bandit property of the hyper model. It is worthy to note that the two ensemble strategies have no parameter to choose. Intuitively, with more trials, the base policies with the bandit property can produce better results as there are more data used for learning. By employing the ensemble strategies that select base policies based on their corresponding overall CTR, we can certainly obtain a unified policy with acceptable CTR. From Table 1, we observe that: (1) The `HyperRandom` policy performs pure exploration on the base policies. This may work well at the beginning of the learning process; however, after a long run, it cannot obtain acceptable performance due to the randomness of the policy selection. (2) The bandit-based ensemble strategies, i.e., `HyperTS` and `HyperTSFB`, are able to achieve comparable performance with the top ranked base policies.

The advantages of the meta-bandit policies involve two aspects: (1) by exploring/exploiting multiple base policies that have different parameter settings, the meta-bandit policies are able to absorb the merits of good policies, and hence produce robust results in terms of the overall CTR; and (2) there is no parameter setting required for meta-bandit policies. `HyperTS` does not have the mechanism of sharing feedbacks among different policies, then for each base policy, the digested click traffic may not be sufficient to produce a high-quality learning model and an accurate CTR estimate. Without enough click traffic, the base policies

Table 1: Relative CTR on the experimental data.

Algorithm	Yahoo! Dataset				KDD Dataset			
	mean	std	min	max	mean	std	min	max
`Random`	1.0000	0.0120	0.9706	1.0149	1.0000	0.0018	0.9972	1.0029
ϵ`-greedy(0.0)`	2.0404	0.0509	1.9527	2.1101	2.3761	0.1398	2.1687	2.6573
ϵ`-greedy(0.01)`	**2.0546**	0.0685	1.9267	2.1776	**2.6204**	0.0266	2.5572	2.6477
ϵ`-greedy(0.1)`	**2.0668**	0.0404	1.9811	2.1114	**2.5282**	0.0107	2.5136	2.5429
ϵ`-greedy(0.3)`	1.8001	0.0303	1.7454	1.8507	2.1974	0.0047	2.1892	2.2041
ϵ`-greedy(0.5)`	1.5265	0.0234	1.4866	1.5550	1.8511	0.0029	1.8470	1.8555
`LinUCB(0.01)`	1.8897	0.0561	1.7832	1.9645	2.3696	0.0885	2.1814	2.4318
`LinUCB(0.1)`	1.3450	0.0169	1.3215	1.3643	2.2158	0.0036	2.2109	2.2227
`LinUCB(0.3)`	1.1961	0.0076	1.1877	1.2118	1.9469	0.0046	1.9404	1.9560
`Softmax(0.01)`	1.9572	0.0572	1.8877	2.0578	**2.5435**	0.0102	2.5196	2.5586
`Softmax(0.1)`	1.1138	0.0133	1.0905	1.1308	1.1946	0.0025	1.1881	1.1971
`Softmax(1.0)`	1.0063	0.0119	0.9809	1.0199	1.0147	0.0012	1.0125	1.0166
`Epoch-greedy(5)`	1.9512	0.0378	1.8750	2.0237	2.3627	0.0064	2.3490	2.3721
`Epoch-greedy(10)`	**2.0517**	0.0841	1.8673	2.1752	**2.5278**	0.0075	2.5141	2.5378
`Epoch-greedy(100)`	**2.0473**	0.0742	1.9198	2.1616	**2.5794**	0.0645	2.4374	2.6379
`Epoch-greedy(500)`	**2.0473**	0.0325	1.9751	2.1022	2.5209	0.0853	2.3721	2.6176
`TS(0.01)`	1.2100	0.0121	1.1909	1.2238	2.0258	0.0021	2.0228	2.0296
`TS(0.1)`	1.1654	0.0074	1.1540	1.1801	1.9715	0.0039	1.9652	1.9794
`TS(1.0)`	1.1401	0.0112	1.1249	1.1603	1.6673	0.0024	1.6625	1.6707
`TS(10.0)`	0.8779	0.0990	0.6907	1.0322	1.2194	0.5090	0.4646	1.8784
`TSNR(0.01)`	1.2728	0.0184	1.2486	1.3060	2.0039	0.0025	2.0003	2.0072
`TSNR(0.1)`	1.2823	0.0108	1.2570	1.2956	2.0589	0.0025	2.0545	2.0621
`TSNR(1.0)`	1.3471	0.0152	1.3208	1.3741	2.2051	0.0019	2.2024	2.2089
`TSNR(10.0)`	1.8847	0.0389	1.8013	1.9270	2.4945	0.0031	2.4882	2.4993
`HyperRandom`	1.1856	0.0099	1.1702	1.2041	1.7379	0.0158	1.7115	1.7597
`HyperTS`	2.0095	0.0708	1.8719	2.1187	2.5175	0.1279	2.2364	2.6587
`HyperTSFB`	**2.1183***	0.0572	2.0115	2.1842	**2.6536***	0.0101	2.6390	2.6709

that exhibit relatively poor performance at the beginning of the trails may have very limited opportunities to be explored/exploited in the subsequent trials, even though they are good policies if running solely. This is the primary reason that the performance of the `HyperTS` policy is inferior to the one of the `HyperTSFB` policy, as indicated in Table 1.

5.4.2 On CTR of Time Buckets

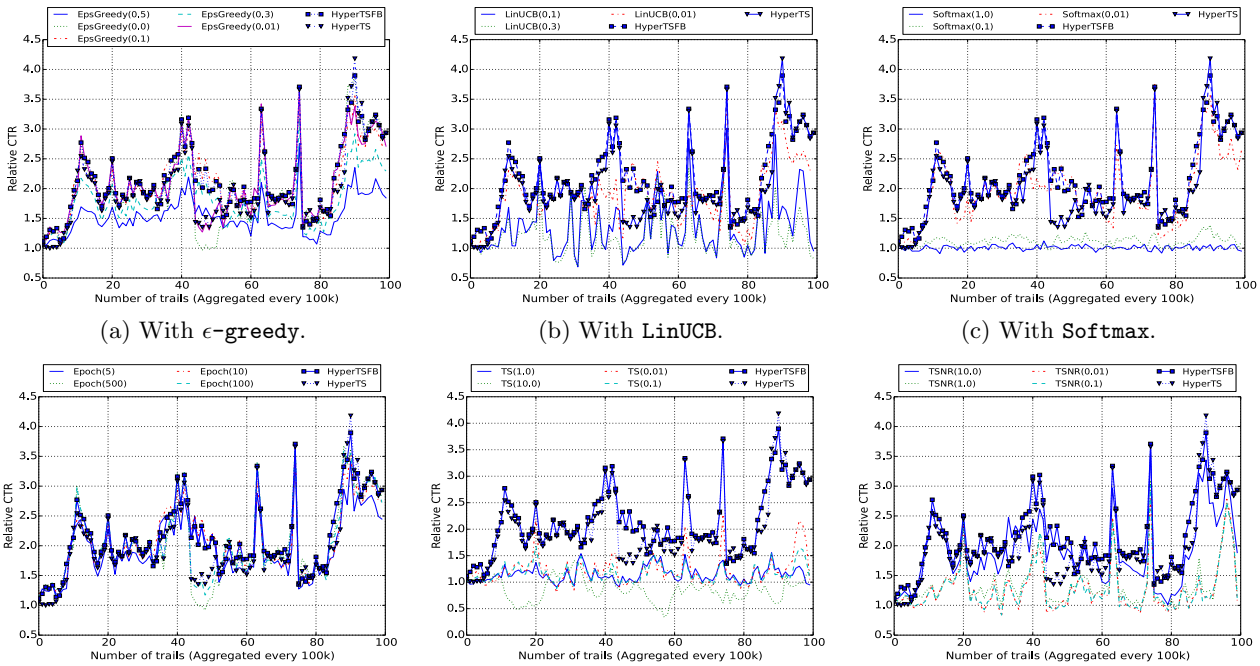
Besides the overall CTR of each policy, we also evaluate the CTR on individual time bucket. The CTR on each bucket is calculated by the clicks collected in that bucket. The entire *Yahoo! Today News* data is split into 100 time buckets, where each bucket has 100,000 impressions on news articles. The *KDD Cup* data set is split into 90 time buckets, with 100,000 impression for each bucket. All the user visit events are order by the time.

For the purpose of illustration, we compare the ensemble strategies, i.e., `HyperTS` and `HyperTSFB`, with each type of contextual bandit policies, including ϵ`-greedy`, `LinUCB`, `Softmax`, `Epoch-greedy`, `TS` and `TSNR`. At each time bucket, these policies are executed independently, and the relative CTR for each policy is calculated. The results for *Yahoo! Today News* data set and *KDD Cup* data set are reported in Figure 1 and 2, respectively. We can observe that the policy of `HyperTSFB` achieves consistent performance on both data sets. Although in some time buckets the relative CTR of `HyperTSFB` is slightly lower than the one of some specific base policies, its overall performance is quite robust compared with other baselines.

From Figure 1 we observe that the CTR curves of different policies have wide fluctuations. This is because the CTR estimated in *Yahoo! news* data is close to the real CTR in each bucket. The real CTR of an online recommender system usually varies over the time. For instance, popular news articles may become unpopular since the news is aging. User interests may change from daytime to nighttime. In contrast, the CTR curves for *KDD Cup* data, as described in Figure 2, are quite flat except the first few time buckets, and the reason is straightforward. The click of *KDD Cup* data is simulated by a group of logistic regression models. The time factor is not included in those models. Despite different characteristics of the two data sets, our proposed meta-bandit policy performs in a very robust way. This further demonstrates the generalization capability of our proposed method in dealing with different recommendation problems.

Another interesting phenomenon is that the CTR lift of `HyperTSFB` is significantly higher than the one of `HyperTS` at the first few time buckets on both *Yahoo! Today News* data and *KDD Cup* data. The reason here is straightforward: by sharing feedbacks among different policies, the data used for exploring/exploiting the policies become rich, and hence there are more data used for training the underlying learning model and estimating the performance of base policies. Therefore, at the initial time buckets, `HyperTSFB` outperforms `HyperTS` in terms of CTR.

To further demonstrate the robustness of our proposed methods, we consider to rank all the policies based on their CTR lift, and then examine if the result of `HyperTSFB` is in the top ranked list. Specifically in each time bucket, we rank the base and ensemble policies based on the CTR lift, and then count the number of times that a policy appears in the top@k ranked list. Next, we calculate the ratio of this count with the total number of buckets for each policy. Finally, we rank the policies based on their ranking ratios. For this evaluation, 4 best performed policies of *Yahoo! Today News* data and *KDD Cup* data are reported in Table 2 and 3, respectively, in which we consider the top@1, top@3 and top@5 results.



(a) With ϵ-greedy. (b) With LinUCB. (c) With Softmax.

(d) With Epoch-greedy. (e) With TS. (f) With TSNR.

Figure 1: Comparison on Yahoo! News Data.

Table 2: CTR ranking on buckets for Yahoo! data.

top@1		top@3		top@5	
Policies	Ratio	Policies	Ratio	Policies	Ratio
HyperTSFB	88.89%	HyperTSFB	100.00%	HyperTSFB	100.00%
ϵ(0.01)	11.11%	ϵ(0.01)	97.78%	ϵ(0.01)	100.00%
-	-	Epoch(100)	78.89%	Epoch(100)	80.00%
-	-	Softmax(0.01)	16.67%	Softmax(0.01)	80.00%

As observed in Table 2, our proposed policy HyperTSFB on *Yahoo! Today News* data always achieves the 1st place in the top@1, top@3 and top@5 results. Also in Table 3, HyperTSFB reaches the 2nd place of top@1, 3rd place of top@3, and 1st place of top@5. The results indicate that HyperTSFB is able to achieve promising performance in most time buckets. Such an observation further confirms the robust capability of our proposed policy in handling different online recommendation problems.

Table 3: CTR ranking on buckets for KDD data.

top@1		top@3		top@5	
Policies	Ratio	Policies	Ratio	Policies	Ratio
ϵ(0.0)	18.00%	ϵ(0.0)	43.00%	HyperTSFB	72.00%
HyperTSFB	12.00%	Epoch(500)	43.00%	Epoch(500)	68.00%
Epoch(500)	12.00%	HyperTSFB	39.00%	ϵ(0.0)	61.00%
LinUCB(0.01)	11.00%	Epoch(100)	34.00%	Epoch(100)	57.00%

In addition, the performance of base policies with different parameter settings may vary significantly over different experimental data. From Table 2, we observe that ϵ-greedy(0.01) and Epoch(100) perform very well on *Yahoo! Today News* data, indicating that for this data set, the bandit policies can have achieve striking performance with a limited exploration. Comparatively in Table 3, ϵ-greedy(0.0) and Epoch(500) are able to produce promising results, meaning that higher CTR can be achieved on *KDD Cup* data based on the policies with much less exploration.

6. CONCLUDING REMARKS

In personalized recommender systems, the dilemma of exploration/exploitation in the cold-start situation remains a challenging issue due to the uncertainty of user preferences. A lot of contextual bandit policies have been proposed to tackle this dilemma; however, the prerequisite of the input parameters limits the predictive power of the policies. In real-world applications, these policies cannot be easily evaluated under different parameters as they may require too much web traffic and affect the profit of service providers.

In this work, we explore ensemble strategies of multiple contextual bandit policies to obtain robust predicted CTR. Specifically, we employ a meta-bandit paradigm that places a hyper bandit over the base bandits, to explicitly explore/exploit the relative importance of base bandits based on user feedbacks. The proposed approach does not have the restriction on the number of policies being involved, and can always obtain an acceptable CTR close the the optimal. Extensive empirical evaluation on two data sets demonstrates the efficacy of our proposed approach in terms of CTR.

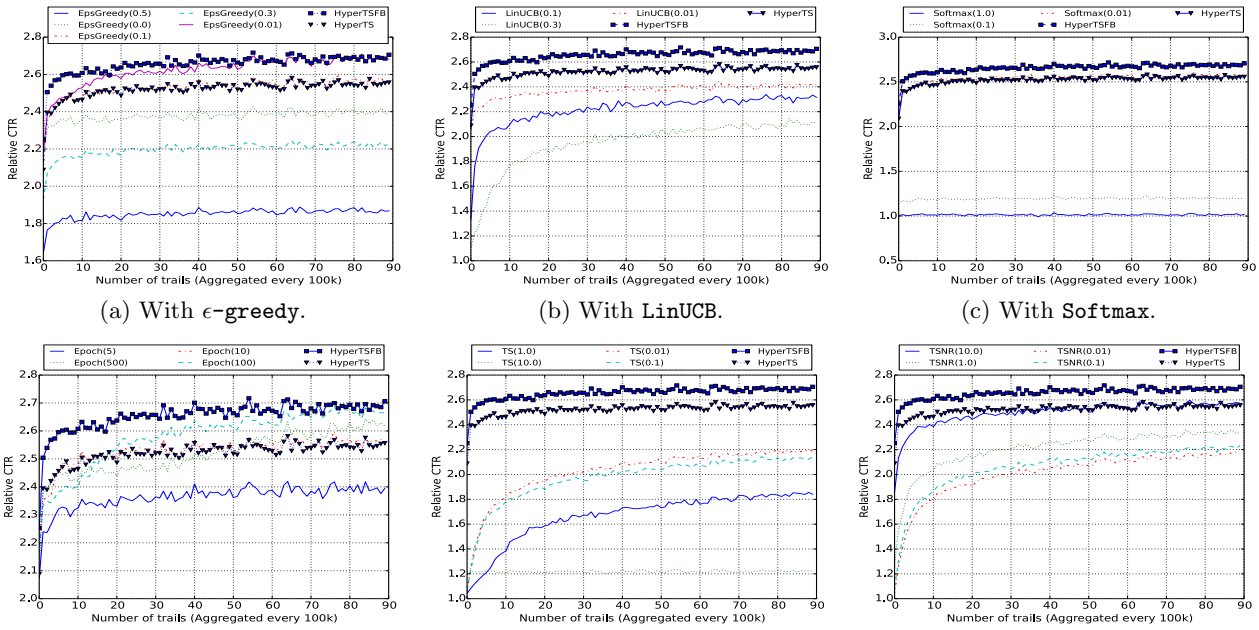
Acknowledgment

The work was supported in part by the National Science Foundation under grants DBI-0850203, HRD-0833093, CNS-1126619, and IIS-1213026, the U.S. Department of Homeland Security under grant number 2010-ST-06200039, Army Research Office under grants W911NF-10-1-0366 and W911NF-12-1-0431, and an FIU Dissertation Year Fellowship.

7. REFERENCES

[1] D. Agarwal et al. Online models for content optimization. In *NIPS*, pages 17–24, 2008.

[2] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002.



(a) With ϵ-`greedy`. (b) With `LinUCB`. (c) With `Softmax`.

(d) With `Epoch-greedy`. (e) With `TS`. (f) With `TSNR`.

Figure 2: Comparison on KDDCup Data.

[3] A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 1998.

[4] C. M. Bishop et al. *Pattern recognition and machine learning*, volume 1. Springer, 2006.

[5] D. Bouneffouf, A. Bouzeghoub, and A. L. Gançarski. A contextual-bandit algorithm for mobile context-aware recommender system. In *NIPS*, pages 324–331, 2012.

[6] O. Chapelle and L. Li. An empirical evaluation of thompson sampling. In *NIPS*, pages 2249–2257, 2011.

[7] M. Gagliolo and J. Schmidhuber. *Algorithm selection as a bandit problem with unbounded losses*. Springer, 2010.

[8] C. Giraud-Carrier. Metalearning-a tutorial. In *ICMLA*, 2008.

[9] R. V. Hogg and E. A. Tanis. *Probability and Statistical Inference*. Prentice Hall, 1996.

[10] M. Jahrer, A. Töscher, and R. Legenstein. Combining predictions for accurate recommender systems. In *SIGKDD*, pages 693–702. ACM, 2010.

[11] Y. Koren. The bellkor solution to the netflix grand prize. *Netflix prize documentation*, 2009.

[12] J. Langford and T. Zhang. The epoch-greedy algorithm for contextual multi-armed bandits. *NIPS*, pages 817–824, 2007.

[13] L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *WWW*, pages 661–670. ACM, 2010.

[14] L. Li, W. Chu, J. Langford, and X. Wang. Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *WSDM*, pages 297–306. ACM, 2011.

[15] F. Maes, L. Wehenkel, and D. Ernst. *Meta-learning of exploration/exploitation strategies: The multi-armed bandit case*. Springer, 2013.

[16] A. Maurer. Algorithmic stability and meta-learning. In *JMLR*, pages 967–994, 2005.

[17] T. G. McKenzie et al. Novel models and ensemble techniques to discriminate favorite items from unrated ones for personalized music recommendation. In *KDDCUP*, 2011.

[18] R. Polikar. Ensemble based systems in decision making. *Circuits and Systems Magazine, IEEE*, 6(3):21–45, 2006.

[19] A. Prodromidis, P. Chan, and S. Stolfo. Meta-learning in distributed data mining systems: Issues and approaches. *Advances in distributed and parallel knowledge discovery*, 3, 2000.

[20] J. B. Schafer, J. A. Konstan, and J. Riedl. Meta-recommendation systems: user-controlled integration of diverse recommendations. In *CIKM*, pages 43–51. ACM, 2002.

[21] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock. Methods and metrics for cold-start recommendations. In *SIGIR*, pages 253–260. ACM, 2002.

[22] S. L. Scott. A modern bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*, 26(6):639–658, 2010.

[23] J. Seiler. *Meta learning in recommendation systems*. Master Thesis, Technical University of Berlin, 2013.

[24] J. Sill, G. Takács, L. Mackey, and D. Lin. Feature-weighted linear stacking. *arXiv preprint arXiv:0911.0460*, 2009.

[25] C. Tekin and M. van der Schaar. Decentralized online big data classification-a bandit framework. *arXiv preprint arXiv:1308.4565*, 2013.

[26] W. R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, pages 285–294, 1933.

[27] M. Tiemann and S. Pauws. Towards ensemble learning for hybrid music recommendation. In *RecSys*, pages 177–178. ACM, 2007.

[28] M. Tokic. Adaptive ε-greedy exploration in reinforcement learning based on value differences. In *KI*, pages 203–210, 2010.

[29] J. Vermorel and M. Mohri. Multi-armed bandit algorithms and empirical evaluation. In *ECML*, pages 437–448, 2005.

[30] M. Wu. Collaborative filtering via ensembles of matrix factorizations. In *KDDCUP*, 2007.

[31] K. Yu, A. Schwaighofer, and V. Tresp. Collaborative ensemble learning: Combining collaborative and content-based information filtering via hierarchical bayes. In *UAI*, pages 616–623, 2002.

[32] B. Zadrozny. Learning and evaluating classifiers under sample selection bias. In *ICML*, pages 114–121. ACM, 2004.

FC-MST: Feature Correlation Maximum Spanning Tree for Multimedia Concept Classification

Hsin-Yu Ha, Shu-Ching Chen
School of Computing and Information Sciences
Florida International University, Miami, FL 33199, USA
{hha001,chens}@cs.fiu.edu

Min Chen
Computing and Software Systems, School of STEM
University of Washington Bothell, WA 98011, USA
minchen2@u.washington.edu

***Abstract*—Feature selection is an actively researched topic in varies domains, mainly owing to its ability in greatly reducing feature space and associated computational time. Given the explosive growth of high-dimensional multimedia data, a well-designed feature selection method can be leveraged in classifying multimedia contents into high-level semantic concepts. In this paper we present a multi-phase feature selection method using maximum spanning tree built from feature correlation among multiple modalities (FC-MST). The method aims to first thoroughly explore not only the correlation between features within and across modalities, but also the association of features towards semantic concepts. Secondly, with the correlations, we identify important features and exclude redundant or irrelevant ones. The proposed method is tested on a well-known benchmark multimedia data set called NUS-WIDE and the experimental results show that it outperforms four well-known feature selection methods in all three important measurement metrics.**

I. Introduction

Feature selection is the process of identifying the most appropriate features from the original feature set based on certain evaluation criteria [1]. It has been intensively explored in various research fields, including pattern recognition [2], [3], machine learning [4], [5], data mining [6]–[8] and statistics [9], to name a few. It is usually applied to reduce high-dimensional feature space by selecting only the relevant and important features. Research shows that a well designed feature selection method can not only handle high-dimensional data sets, but also successfully enhance classification performance in coping with imbalanced data where one class has way more data instances than the other class(es) [5], [10]–[13]. Hence, feature selection has been widely applied in applications with imbalanced datasets such as medical decision making using MRI images [14] or EMG signals [15], biomedical studies using microarray gene data sets [16], and text categorization [11], [17], etc.

Generally speaking, feature selection methods can be categorized into three classes, supervised algorithms [18], [19], unsupervised algorithms [20], [21], or semi-supervised algorithms [22], [23]. As supervised algorithms require a set of labeled training data that generally involves expensive human labor, many researchers are increasingly focused on unsupervised or semi-supervised methods in selecting good features. On the other hand, feature selection methods can also be classied into different types of strategies including filter, wrapper, and embedded methods [9]. In filter methods [24], only the general characteristics of training data are considered to evaluate the predefined relevance score of each feature. No learning algorithms or induction algorithms are involved during the process. Therefore, it has a lower computational cost compared to the other two. The wrapper methods [25] work closely with certain classification algorithm whose classification results are used as the evaluation criteria to determine whether a subset of features captures relevant information. The feature subset produces the least classification errors will be selected to build the classification model. Usually, the wrapper methods can outperform the filter methods with regard to classification accuracy. However, the process requires a proper integration of multiple components including a predefined classification algorithm, a good feature relevance criterion, and an efficient searching method to identify feature subset. In addition, it is computationally intensive and may lead to over-fitting problem. Lastly, the embedded methods [26], [27] incorporate learning methods by using objective functions to evaluate feature relevance and select relevant feature subset. Unlike wrapper methods, it doesn't search through the space of all possible feature subsets but identify feature subsets via selected learning strategy. Hence, it is less computationally expensive. In addition, it is also less prone to overfitting compared to wrapper methods.

In this work, we propose a feature selection method called FC-MST to cope with high-dimensionalities and imbalanced problem in multimedia concept detection. The proposed method first applied Multiple Correspondence Analysis (MCA) to project original features into a two-dimensional feature space and obtain feature correlations. Then, a Maximum Spanning Tree is built using the correlations and eliminate irrelevant and redundant features by pruning the tree. The goal is to explore possible feature correlations within and among different modalities and further utilize the correlation to identify the ones that are important and highly relevant to the targeted semantic concepts.

The rest of the paper is organized as follows. We present the overview of the proposed framework and the detail of each component in section 2. In section 3, we explain the design of the experiments and analyze the experimental results. Finally, the paper is concluded in section 4.

II. Proposed Framework

For each semantic concept, the proposed FC-MST feature selection method aims to identify a feature subset, containing the important and relevant features from the original multi-modal feature set, to improve the performance of semantic concept classification. It is a three-step supervised method as shown in Figure 1.

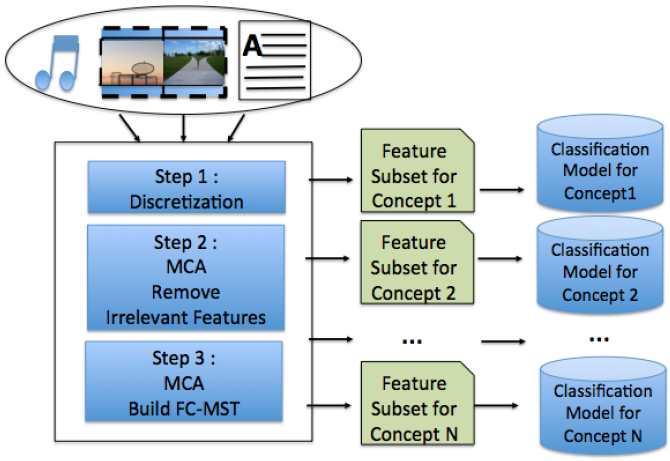


Fig. 1: An overview of the proposed framework

A. Step1: Features Eliminated from Discretization Process

To handle both numeric and nominal features, a supervised method called Minimum Description Length (MDL) [28] is used to discretize each feature into a number of intervals based on its values associated with a target concept. For example, Table I shows 5 instances with M features and two columns at the end indicates the label of positive or negative concept. If an instance has value 1 in the positive concept column, it means the concept can be observed from the instance, and vice versa.

TABLE I: Example of the Original Features

	Feature 1	Feature 2	...	Feature M	Target Concept Positive	Target Concept Negative
Inst. 1	-0.49	1.08	...	-0.45	1	0
Inst. 2	-0.56	-0.85	...	-1.32	0	1
Inst. 3	-0.61	-2.21	...	1.33	1	0
Inst. 4	-0.48	-0.97	...	-1.01	0	1
Inst. 5	-0.53	-1.54	...	0.97	1	0

After discretization, all feature values are grouped into intervals and are denoted as F_j^i where i is the index of feature and j is the index of the interval. For instance, F_3^2 means the third interval of the second feature. Table II shows example discretization results of Table I. As we can see, all instances share the same value in the feature 1 column (i.e., F_1^1). This means feature 1 doesn't have the distinguish ability for the target concept and such features will be removed in the first step of our proposed method as shown in Algorithm 1.

TABLE II: Example of the Discretized Features

	Feature 1	Feature 2	...	Feature M	Target Concept Positive	Target Concept Negative
Inst. 1	F_1^1	F_3^2	...	F_2^M	1	0
Inst. 2	F_1^1	F_2^2	...	F_1^M	0	1
Inst. 3	F_1^1	F_1^2	...	F_3^M	1	0
Inst. 4	F_1^1	F_3^2	...	F_1^M	0	1
Inst. 5	F_1^1	F_1^2	...	F_3^M	1	0

Algorithm 1: Feature eliminated from discretization process

input : The given training data set D with feature set as $TDF = F_1, F_2, ..., F_M$, along with the class label C

output: SF_1: A set of selected features

```
SF_1 ⟵ TDF;
for i ← 1 to M do
    NumFI_i = |MDL(F_i)|;
    /* NumFI_i represents the number of
       intervals in the i^th feature      */
    if NumFI_i = 1 then
        SF_1 ⟵ SF_1 − {F_i};
    end
end
return SF_1
```

B. Step2 : Features Eliminated from MCA

Multiple Correspondence Analysis (MCA) has been applied and proven effective to the research areas ranging from feature selection [29], discretization [30], video semantic concept detection [31]–[38], to data pruning [39]. In this paper, our previous work [29] is integrated as a preprocess step, which has been demonstrated to outperforms other existing feature selection methods, such as information gain (IG), Chi-Square measure (CHI), etc., in terms of classification accuracy.

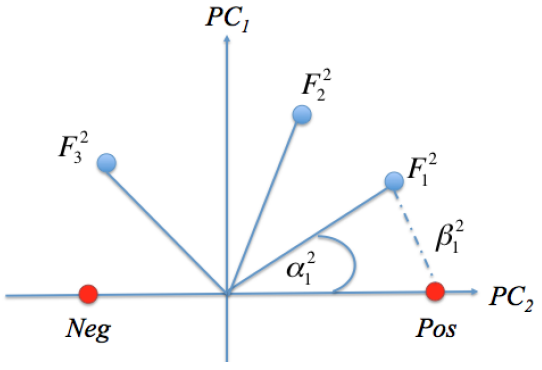


Fig. 2

After applying MCA to a data set as presented in Table II,

Algorithm 2: Features Eliminated from MCA

input : A given training data set D_1 with selected feature set $SF_1 = F_1, F_2, ..., F_L$, along with the class label C

output: SF_2: A set of selected features

$SF_2 \longleftarrow SF_1$;
for $i \leftarrow 1$ **to** L **do**
  $(FIC, FIR) = MCA(D_1)$;
  /* Correlation and reliability of feature interval toward target concept */
  for $j \leftarrow 1$ **to** $NumFI_i$ **do**
    $SumCorrelation+ = FIC_j$;
    $SumReliability+ = FIR_j$;
  end
  $FC_i = (SumCorrelation + SumReliability)/NumFI_i$
end
if $FC_i < TH$ **then**
  $SF_2 \longleftarrow SF_1 - \{F_i\}$;
end
return SF_2

all the intervals of a feature are projected on a two-dimensional space composed by two major principal components, PC_1 and PC_2. Figure 2 depicts three intervals of feature 2 and two red dots which represent positive and negative classes. The relation between an interval of a particular feature and the positive class can be represented by two factors. One is called *Correlation* α_j^i (e.g., α_1^2), which is the cosine value of the angle between the feature interval F_j^i (e.g., F_1^2) and the positive class. The other is called *Reliability* β_j^i (e.g., β_1^2), which is the distance between a feature interval F_j^i (e.g., F_1^2) and the positive class. Together these two can be used as a relevance score of a feature interval toward the semantic concept. Zhu et al. [29] go further to obtain the average relevance score per feature to eliminate features whose score is lower than a preset threshold as shown in Algorithm 2. This method is adopted here as a preprocess step to obtain important features for building Maximum Spanning Tree (MST) in step 3.

C. Step3 : Feature Eliminated from FC-MST

1) Building Feature Correlation Adjacency Matrix: In section II-B, MCA is used to capture correlation between feature intervals and the positive target concept as shown in Figure 2. To build the maximum spanning tree, we apply MCA to the remaining features from section II-B to explore correlations between each pair of them. Take Figure 3 as an example, all the intervals of the second feature F^2 and the third feature F^3 are projected onto the two-dimensional symmetric map. The cosine value of each pair of intervals from different features will be generated and the maximum value is selected as the correlation between this pair of features as shown in equation

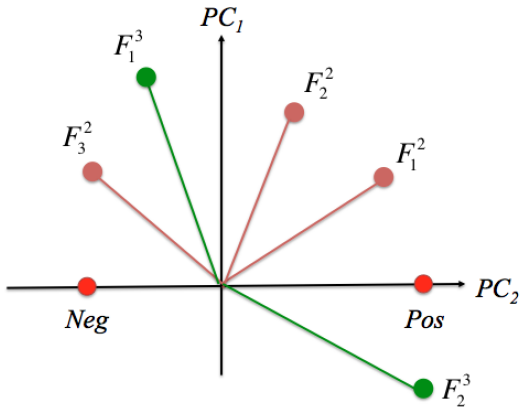


Fig. 3

Algorithm 3: Building Feature Correlation Adjacency Matrix

input : A given training data set D_2 with selected feature set $SF_2 = F_1, F_2, ..., F_L$, along with the class label C

output: Adjacency Matrix AM and the corresponding undirected weighted graph $G(F, E)$

for $i \leftarrow 1$ **to** L **do**
  for $j \leftarrow 1$ **to** L **do**
    $(FIC, FIR)_{ij} = MCA(D_1)$;
    /* Correlation and reliability of feature intervals of one feature toward feature intervals of the other feature */
    if $i = j$ **then**
      $AM(i, j)$ = 0 ;
    else
      $AM(i, j) = Max(FIC, FIR)_{ij}$;
    end
  end
end
return AM

1.

$$FC_{ij} = \begin{cases} \text{argmax}\, Cos(\alpha_{F_m^i F_n^j}), & \text{if } i \neq j \\ 0, & \text{if } i = j \end{cases} \quad (1)$$

Here, i and j are indexed from 1 to L, the total number of the remaining features. The feature correlation between any feature and itself is set to be zero. Therefore, an $L*L$ adjacent matrix can be obtained where each feature is a vertex and the correlation is the edge. Consecutively, an undirected weighted graph $G(F, E)$ is built upon the adjacent matrix where F is the set of remaining features and E indicates the set of feature correlation $\{FC_{ij}\}_{i,j=1}^{L}, i \neq j$.

Algorithm 4: FC-MST
Feature Correlaion Maximum Spanning Tree

input : An undirected weighted graph $G(F, E)$, comprising a set of features SF_2 = F_1, F_2, ..., F_L together with a set of edges which have feature correlation between each feature pairs as the value FC_ij where i and $j \in 1, 2, ...L$, $i < j$. A set of Feature Correlation toward target concept FC_iC where $i \in 1, 2, ...L$

output: SF_3: A set of selected features

1 $SF_3 \longleftarrow \emptyset$; /* Selected features starts with an empty set */
2 $MaxSpanTree = Prim(G)$; /* Applying Prim algorithm on undirected weighted graph G */
3 **for** *each Edge* $E_{ij} \in MaxSpanTree$ **do**
4 | **if** $FC_{ij} < FC_{iC}$ **and** $FC_{ij} < FC_{jC}$ **then**
5 | | $MaxSpanTree \longleftarrow MaxSpanTree - E_{ij}$
6 | **end**
7 **end**
8 $C = BreadFirstSearch(MaxSpanTree)$;
/* Apply BFS algorithm and return a set of components */
9 **for** *Each Component* $C_m \in C$ **do**
10 | $SF_3 \longleftarrow MaxFC(C_m)$
11 **end**
12 return SF_3

2) Building Feature Correlation Maximum Spanning Tree: There are three purposes of building a feature correlation maximum spanning tree as listed below:

- Partition FC-MST into relevant feature clusters which have high intra-cluster correlation and low inter-cluster correlation
- Identify representative features from each feature clusters
- Eliminate redundant and irrelevant features from FC-MST

As shown in Algorithm 4, given the undirected weighted graph from section II-C, a maximum spanning tree is constructed using Prim's method [40] which spans over all the feature vertices based on the correlation values. In brief, the proposed FC-MST is an acyclic subgraph that has the maximum sum of feature correlation weights across all the features nodes. Once the maximum spanning tree is built, the proposed algorithm (see statement 2 in Algorithm 4) loops through all the edges and removes the ones whose weight FC_{ij} is smaller than the correlation of features toward concept, e.g., FC_{iC} and FC_{jC} (see statements 3 to 7 in Algorithm 4). Breadth-first search (BFS) [41] is applied to identify a set of disconnected components (i.e., clusters) $C = C_1, C_2, ..., C_N$ after such edges removal. The feature with the largest correlation toward the target concept in one cluster will be selected as its representative feature. Since every cluster is composed by highly correlated features, all the other features besides the representative one are considered redundant and they are removed from the feature set (see statements 8 to 11 in Algorithm 4). At the end, a subset of representative features is selected to build the classification model for each semantic concept.

III. Experiments

A. Dataset

NUS-WIDE [42], a large-scale image data set containing 269,648 images and the associated tags, is introduced to evaluate the performance of the proposed feature selection method. It has six types of low-level visual features extracted from the images, e.g., color histogram, color correlogram, edge direction histogram, etc., and user tags from flickr website represented as text features. There are 81 high-level semantic concepts, most of them highly imbalanced with the PN ratio (i.e., the number of positive instances vs. negative ones) lower than 1%.

B. Evaluation Criteria

As discussed earlier, a general use of the feature selection method is to identify a subset of representative features that enable classifiers to build better classification models more efficiently. Therefore, we can assess the performance of a feature selection method by evaluating performance of the resulting classification model and efficiency of the classification process. Consequently, the proposed feature selection method is evaluated and compared with other state-of-the-art methods using three criteria.

1) **Classification Model Performance**

TABLE III: Confusion Matrix

		Predicted Class	
		Positive	**Negative**
Actual Class	**Positive**	TruePos	FalseNeg
	Negative	FalsePos	TrueNeg

Confusion matrix (see an example in Table III) is widely used in machine learning and data mining areas to visu-

alize classification results in table-layout fashion. Many performance metrics can be derived from it to analyze the classification results from different perspectives.

- **Precision**
  Based on Table III, precision is calculated as

$$Precision = \frac{TruePos}{(TruePos + FalsePos)} \quad (2)$$

  In other words, precision shows the fraction of retrieved instances that are relevant, where a high precision indicates a lower false positive rate.
- **Average Precision and Mean Average Precision**
  Average precision (AP) and mean average precision (MAP) are two metrics extended from precision, as defined in equation 3 and equation 4, respectively. In brief, **Average Precision** at K is used to evaluate top K ranked results, where $\#(TopR)$ represents the number of instances which are correctly classified as positive instances among top R retrieved instances, $R = 1...K$. A high AP value means more relevant results are ranked earlier than irrelevant ones.

$$AP(K) = \frac{1}{K} \sum_{R=1}^{K} \frac{\#(TopR)}{R} \quad (3)$$

  Mean Average Precision is used to validate ranked results for more than one concepts, where TC is the total number of concepts and $AP_C(K)$ is the average precision at K for concept C.

$$MAP(K) = \frac{\sum_{C=1}^{TC} AP_C(K)}{TC} \quad (4)$$

2) **Feature Reduction Rate** The purpose of feature selection method is to select the most relevant and important features while greatly reducing the feature space. Hence, the proposed method is also evaluated in terms of feature reduction rate, which is calculated in equation 5.

$$FRR = \frac{(OF\# - FS\#)}{OF\#} \quad (5)$$

where $OF\#$ represents the number of original features and $FS\#$ represents the number of remaining features after applying feature selection method.

3) **Efficiency Rate** Lastly, efficiency rate is defined by taking both MAP value and processing time into account as shown in equation 6.

$$ER = \frac{MAP(K)}{ProcessingTime} \quad (6)$$

On one hand, a higher MAP value indicates more positive instances being successfully given higher ranking scores. On the other hand, a reduced feature space leads to shorter processing time. Therefore, given the equation 6, a higher efficiency Rate (ER) represents a better overall performance for a feature selection method.

C. Experimental results

In the experiments, our proposed method is compared with four well-known feature selection methods, e.g., ChiSquare, Filter, InfoGain, and Wrapper. After feature selection on the NUS-WIDE data, Support vector machine (SVM), a constructive learning algorithm, is used to build classification models. SVM is chosen because of its capability in classifying high-dimensional data [43]. Three-fold cross validation scheme is adopted to avoid bias.

First, the experimental result demonstrates the comparison between the proposed method and the other feature selection methods in terms of the MAP values. As shown in Table IV, the proposed method FC-MST achieves the highest MAP values and thus outperforms all other methods in all cases, where K is set to different values in the range of 5 to 200. The proposed method is also the only feature selection method that maintains over 0.7 MAP value across all cases. The trend can also be seen in Figure 4.

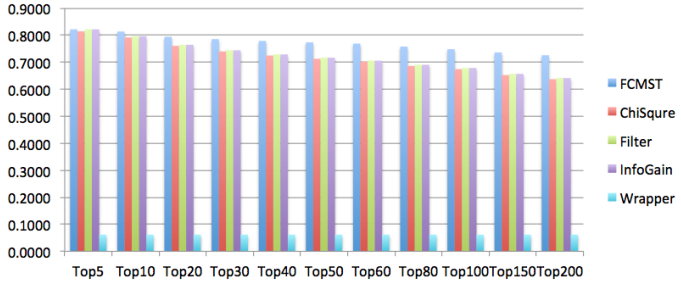


Fig. 4: The MAP values of 81 concepts in NUS-WIDE for different retrieved levels against other feature selection methods

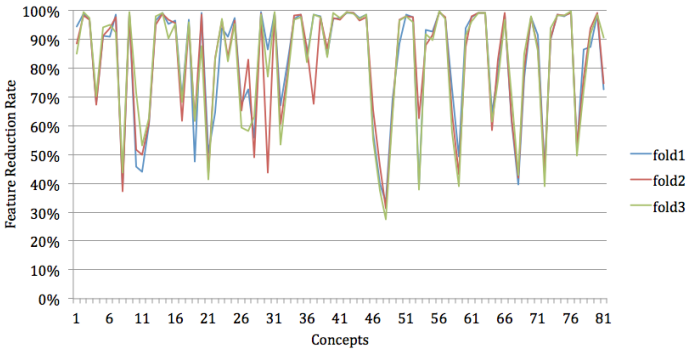


Fig. 5: Feature Reduction Rate (FRR) for NUS-WIDE 81 concepts after applying FC-MST

Secondly, Figure 5 depicts the feature reduction rate (FRR) over all 81 concepts after applying the proposed feature selection method. Among them, we achieved more than 90% FRRs on 40 concepts. The experiment indicates that the proposed method can greatly reduce the original feature space and are especially helpful in dealing with high-dimensional data sets.

TABLE IV: The MAP values of 81 concepts in NUS-WIDE against other feature selection methods

Method	K = 5	K = 10	K = 20	K = 30	K = 40	K = 50	K = 60	K = 80	K = 100	K = 150	K = 200
FC-MST	**0.8217**	**0.8133**	**0.7940**	**0.7854**	**0.7786**	**0.7734**	**0.7688**	**0.7578**	**0.7481**	**0.7361**	**0.7257**
ChiSqure	0.8140	0.7917	0.7604	0.7398	0.7246	0.7125	0.7015	0.6862	0.6744	0.6524	0.6370
Filter	0.8215	0.7961	0.7645	0.7439	0.7287	0.7166	0.7057	0.6903	0.6785	0.6566	0.6412
InfoGain	0.8215	0.7961	0.7645	0.7439	0.7287	0.7166	0.7056	0.6903	0.6785	0.6566	0.6411
Wrapper	0.0617	0.0617	0.0617	0.0617	0.0617	0.0617	0.0617	0.0617	0.0617	0.0617	0.0617

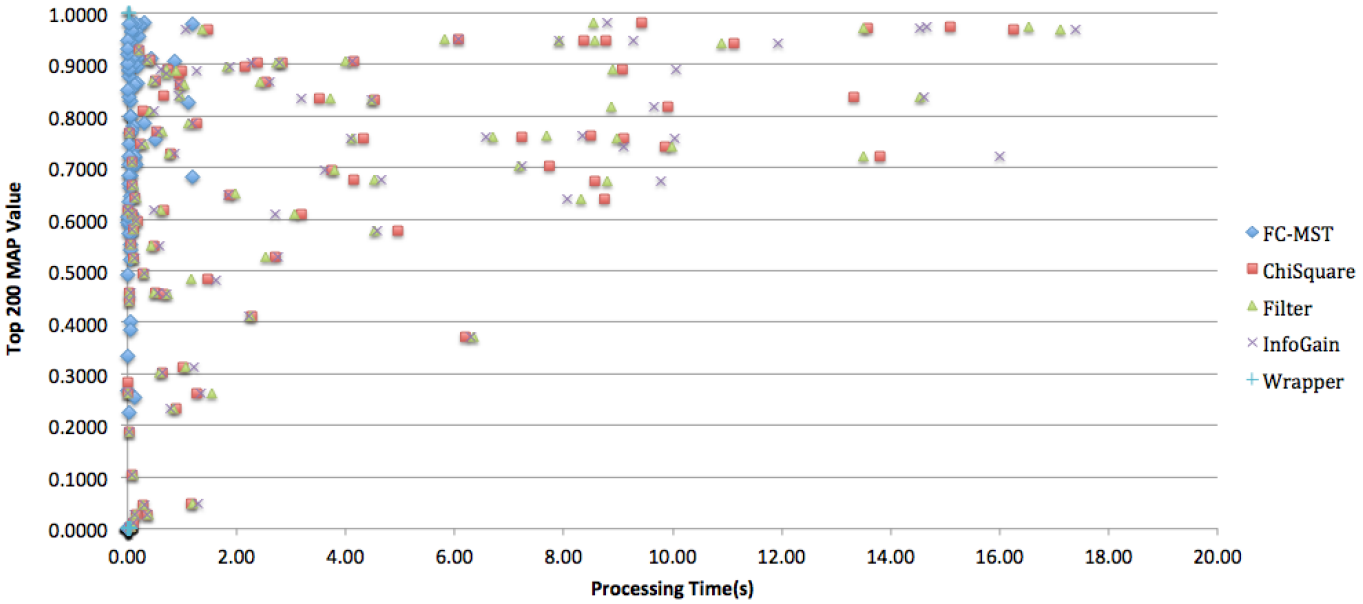


Fig. 6: Top200 Map Value v.s. Processing Time against other feature selection methods

Thirdly, experiment is conducted to validate whether the proposed method is able to reduce the processing time meanwhile producing a compatible classification results against other methods in terms of MAP value. In Figure 6, the results are projected on a two-dimensional chart, where x-axis represents the computation time for the classification process in seconds and y-axis shows the MAP values at K = 200. As shown in Figure 6, the proposed FC-MST method can achieve similar or better MAP value as compared to other methods while using significantly shorter processing time.

Lastly, the efficiency rate is calculated as defined in equation 6 using MAP value at K = 200. In Figure 7, it can be easily observed that FC-MST has the highest efficiency rate across all the 81 concepts except for a few concepts where the wrapper method produces better rates. This is because the wrapper method selects only one feature, its processing time is the shortest. However, as can be seen in Table IV, the wrapper method produces much worse MAP values (always the worst among all methods).

IV. Conclusion

In this paper, we propose a three-steps feature selection method FC-MST. It uses Multiple Correspondence Analysis to explore correlation among features within and across modalities and to capture correlation between feature and targeted semantic concepts. It also allows visual depicts of feature correlation using Maximum Spanning Tree. Consequently, it enhances the classification performance on multimedia data by effectively removing redundant and irrelevant features from high-dimensional data. As shown in the experiments, FC-MST outperforms four other well-known feature selection methods in all three perspectives: MAP, feature reduction rate, and efficient rate. It proves that the proposed method can not only greatly reduce computational cost owing to feature space reduction, but also lead to better classification results.

Acknowledgment

This research was supported in part by the U.S. Department of Homeland Security under grant Award Number 2010-ST-062-000039, the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0001, and NSF HRD-0833093.

References

[1] L. Liu, J. Kang, J. Yu, and Z. Wang, "A comparative study on unsupervised feature selection methods for text clustering," in *Proceedings of 2005 IEEE International Conference on Natural Language Processing and Knowledge Engineering*. IEEE, 2005, pp. 597–601.

[2] A. Jain and D. Zongker, "Feature selection: Evaluation, application, and small sample performance," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, no. 2, pp. 153–158, 1997.

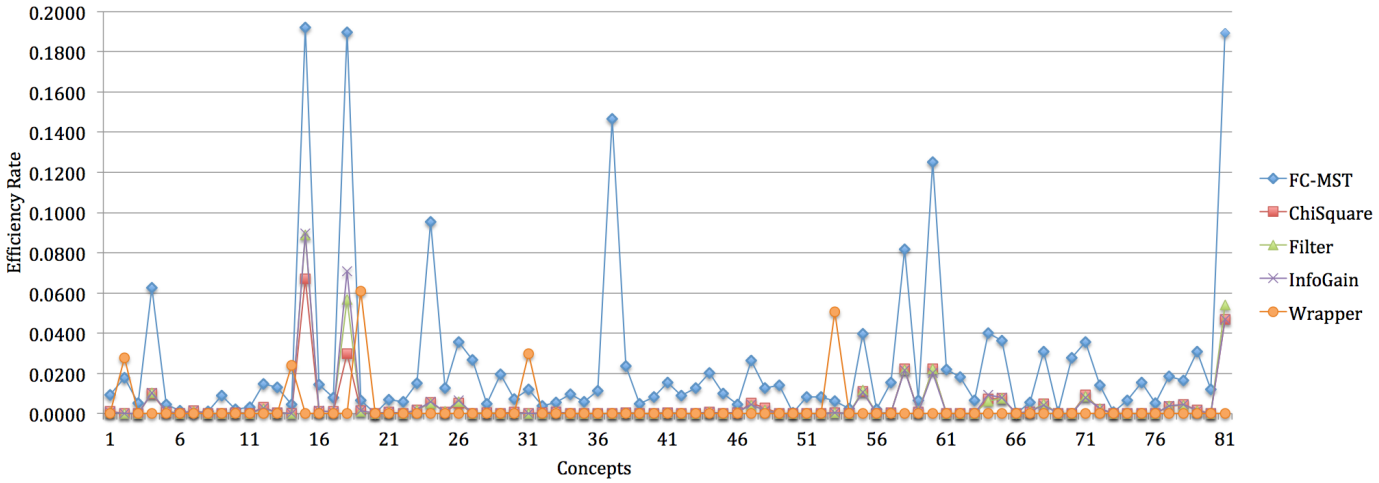


Fig. 7: The efficiency rate of 81 concepts in NUS-WIDE against other feature selection methods

[3] K. Fukunaga, *Introduction to statistical pattern recognition*. Academic press, 1990.

[4] M. A. Hall, "Correlation-based feature selection for machine learning," Ph.D. dissertation, The University of Waikato, 1999.

[5] D. Mladenic and M. Grobelnik, "Feature selection for unbalanced class distribution and naive bayes," in *Proceedings of the 16th International Conference on Machine Learning (ICML*. Citeseer, 1999.

[6] H. Liu and H. Motoda, *Feature selection for knowledge discovery and data mining*. Springer, 1998.

[7] L. Yu and H. Liu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," in *ICML*, vol. 3, 2003, pp. 856–863.

[8] M. Chen, S.-C. Chen, and M.-L. Shyu, "Hierarchical temporal association mining for video event detection in video databases," in *2007 IEEE 23rd International Conference On Data Engineering Workshop*. IEEE, 2007, pp. 137–145.

[9] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *The Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[10] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *The Journal of machine learning research*, vol. 3, pp. 1289–1305, 2003.

[11] Z. Zheng, X. Wu, and R. Srihari, "Feature selection for text categorization on imbalanced data," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 80–89, 2004.

[12] X.-w. Chen and M. Wasikowski, "Fast: a roc-based feature selection metric for small samples and imbalanced data classification problems," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2008, pp. 124–132.

[13] S.-C. Chen, R. L. Kashyap, and A. Ghafoor, *Semantic Models for Multimedia Database Searching and Browsing*. Springer, 2000, vol. 21.

[14] N. Zhang, S. Ruan, S. Lebonvallet, Q. Liao, and Y. Zhu, "Kernel feature selection to fuse multi-spectral mri images for brain tumor segmentation," *Computer Vision and Image Understanding*, vol. 115, no. 2, pp. 256–269, 2011.

[15] A. Phinyomark, P. Phukpattaranont, and C. Limsakul, "Feature reduction and selection for emg signal classification," *Expert Systems with Applications*, vol. 39, no. 8, pp. 7420–7431, 2012.

[16] C.-P. Lee and Y. Leu, "A novel hybrid feature selection method for microarray data analysis," *Applied Soft Computing*, vol. 11, no. 1, pp. 208–213, 2011.

[17] H. Uğuz, "A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm," *Knowledge-Based Systems*, vol. 24, no. 7, pp. 1024–1032, 2011.

[18] L. Song, A. Smola, A. Gretton, J. Bedo, and K. Borgwardt, "Feature selection via dependence maximization," *The Journal of Machine Learning Research*, vol. 98888, no. 1, pp. 1393–1434, 2012.

[19] J. Weston, A. Elisseeff, B. Schölkopf, and M. Tipping, "Use of the zero norm with linear models and kernel methods," *The Journal of Machine Learning Research*, vol. 3, pp. 1439–1461, 2003.

[20] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Advances in neural information processing systems*, 2005, pp. 507–514.

[21] J. G. Dy and C. E. Brodley, "Feature selection for unsupervised learning," *The Journal of Machine Learning Research*, vol. 5, pp. 845–889, 2004.

[22] Z. Xu, I. King, M.-T. Lyu, and R. Jin, "Discriminative semi-supervised feature selection via manifold regularization," *Proceedings of IEEE Transactions on Neural Networks*, vol. 21, no. 7, pp. 1033–1047, 2010.

[23] Z. Zhao and H. Liu, "Semi-supervised feature selection via spectral analysis." in *SDM*. SIAM, 2007, pp. 641–646.

[24] A. L. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artificial intelligence*, vol. 97, no. 1, pp. 245–271, 1997.

[25] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial intelligence*, vol. 97, no. 1, pp. 273–324, 1997.

[26] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani *et al.*, "Least angle regression," *The Annals of statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[27] J. R. Quinlan, *C4. 5: programs for machine learning*. Morgan kaufmann, 1993, vol. 1.

[28] U. M. Fayyad and K. B. Irani, "On the handling of continuous-valued attributes in decision tree generation," *Machine learning*, vol. 8, no. 1, pp. 87–102, 1992.

[29] Q. Zhu, L. Lin, M.-L. Shyu, and S.-C. Chen, "Feature selection using correlation and reliability based scoring metric for video semantic detection," in *Proceedings of 2010 IEEE Fourth International Conference on Semantic Computing (ICSC)*. IEEE, 2010, pp. 462–469.

[30] Q. Zhu, L. Lin, M.-L. Shyu, and S.-C. Chen, "Effective supervised discretization for classification based on correlation maximization," in *Proceedings of 2011 IEEE International Conference on Information Reuse and Integration (IRI)*. IEEE, 2011, pp. 390–395.

[31] L. Lin, M.-L. Shyu, and S.-C. Chen, "Association rule mining with a correlation–based interestingness measure for video semantic concept detection," *International Journal of Information and Decision Sciences*, vol. 4, no. 2, pp. 199–216, 2012.

[32] L. Lin and M.-L. Shyu, "Effective and efficient video high-level semantic retrieval using associations and correlations," *International Journal of Semantic Computing*, vol. 3, no. 04, pp. 421–444, 2009.

[33] L. Lin and M.-L. Shyu, "Weighted association rule mining for video semantic detection," *International Journal of Multimedia Data Engineering and Management (IJMDEM)*, vol. 1, no. 1, pp. 37–54, 2010.

[34] L. Lin, G. Ravitz, M.-L. Shyu, and S.-C. Chen, "Correlation-based video semantic concept detection using multiple correspondence analysis," in *Proceedings of the Tenth IEEE International Symposium on Multimedia, 2008. ISM'09*. IEEE, 2008, pp. 316–321.

[35] L. Lin, G. Ravitz, M.-L. Shyu, and S.-C. Chen, "Effective feature space reduction with imbalanced data for semantic concept detection," in *2008*

IEEE on Sensor Networks, Ubiquitous and Trustworthy Computing, SUTC'08.International Conference on. IEEE, 2008, pp. 262–269.

[36] L. Lin, G. Ravitz, M.-L. Shyu, and S.-C. Chen, "Video semantic concept discovery using multimodal-based association classification," in *2007 IEEE International Conference on Multimedia and Expo*. IEEE, 2007, pp. 859–862.

[37] L. Lin, C. Chen, M.-L. Shyu, and S.-C. Chen, "Weighted subspace filtering and ranking algorithms for video concept retrieval," *IEEE on MultiMedia*, vol. 18, no. 3, pp. 32–43, 2011.

[38] M.-L. Shyu, Z. Xie, M. Chen, and S.-C. Chen, "Video semantic event/concept detection using a subspace-based multimedia data mining framework," *IEEE Transactions on Multimedia*, vol. 10, no. 2, pp. 252–259, 2008.

[39] L. Lin, M.-L. Shyu, and S.-C. Chen, "Enhancing concept detection by pruning data with mca-based transaction weights," in *Proceedings of the 11th IEEE International Symposium on Multimedia, 2009. ISM'09.* IEEE, 2009, pp. 304–311.

[40] R. C. Prim, "Shortest connection networks and some generalizations," *Bell system technical journal*, vol. 36, no. 6, pp. 1389–1401, 1957.

[41] E. F. Moore, *The shortest path through a maze*. Bell Telephone System., 1959.

[42] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, "Nus-wide: a real-world web image database from national university of singapore," in *Proceedings of the ACM international conference on image and video retrieval*. ACM, 2009, p. 48.

[43] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

Optimizing Online Spatial Data Analysis with Sequential Query Patterns

Chunqiu Zeng, Hongtai Li, Huibo Wang, Yudong Guang,
Chang Liu, Tao Li, Mingjin Zhang, Shu-Ching Chen, Naphtali Rishe
School of Computing and Information Sciences,
Florida International University
{czeng001, hli019, hwang033, yguan004, cliu019, taoli, zhangm, chens, rishen}@cs.fiu.edu

***Abstract*—The exponential growth of the usage of geographic information services leads to an extensive popularity on spatial data analysis in both industrial and academic communities. However, it is quite challenging for users to efficiently analyze and quickly understand the spatial data due to the inherently complex and dynamic nature of GIS applications.**

To address the challenges, this paper presents an approach to optimize the online spatial data analysis by mining the sequential query patterns from the user query logs of GIS applications. The sequential query patterns are used to automatically generate the query template, from which the users are able to quickly compose new queries. The sequential query patterns contribute to the workflow construction for complex spatial data analysis tasks as well. Our proposed approach takes advantage of the generated workflow to parallelize the independent spatial analysis tasks. As a result, the throughput of our system has been increased greatly and more efficient geographic information services are made available to the users. We present a case study to demonstrate the efficiency and effectiveness of the proposed approach by integrating two software systems at Florida International University (FIU): TerraFly (an GIS application) and FIU-Miner (a distributed data mining framework).

I. Introduction

With the rapid advancement in technology of geographic information system, online spatial data analysis becomes increasingly essential in various application domains such as water management, crime mapping, disease analysis, and real estate. As a consequence, many geographic applications from different domains emerge recently in the form of web applications or mobile applications. Miscellaneous requirements from diverse application domains strongly dictate efficient support for spatial data analysis.

However, the inherently complex and dynamic nature of GIS applications gives rise to great challenges for users to efficiently analyze and quickly understand the spatial data. Spatial data analysis conducted on a typical geographic application tends to involve a series of complicated interactions and may be fussy with a lot of low-level details. In addition, users from different domains want GIS systems to dynamically create map applications on their own spatial data sets. Moreover, massive spatial data analysis conducted on GIS applications is very resource-consuming. These big challenges require many GIS applications to be designed with innovative approaches to gain competitive advantages.

Recently, TerraFly GeoCloud is designed and developed to support spatial data analysis and visualization [7]. Point and polygon spatial data can be accurately visualized and manipulated in TerraFly GeoClould. It allows users to visualize and share spatial data related to different domains such as real property, crime, and water resources. Online analysis of spatial data is supported by the spatial data analysis engine of TerraFly GeoCloud as well. In order to efficiently support complex spatial data analysis and flexible visualization of the analysis results, MapQL, a SQL-like language, is implemented to represent the analysis queries in TerraFly GeoClould. A MapQL statement is capable of defining an analysis task and customizing the visualization of analysis results. According to the queries, the spatial data analysis engine completes the analysis task and renders the customized visualization of analysis results. For instance, given the real property data, a user may want to explore the house prices near Florida International University. The corresponding MapQL statement for such an exploration is shown in Figure 1.

```
SELECT
        '/var/www/cgi-bin/house.png' AS T_ICON_PATH,
        r.price AS T_LABEL,
        '15' AS T_LABEL_SIZE,
        r.geo AS GEO
FROM
        realtor_20121116 r
WHERE
        ST_Distance(r.geo, GeoFromText('POINT(-80.27, 25.757228)')) < 0.3;
```

Fig. 1: A MapQL query on real property data is given, where POINT(-80.27,25.757228) is the location of Florida International University.

A MapQL statement extends the semantics of traditional SQL statements by introducing new reserved key words. As shown in Figure 1, *T_ICON_PATH*, *T_LABEL*, *T_LABEL_SIZE* and *GEO* are four additional reserved words in a MapQL statement. These four reserved key words are used in the "$expression\ AS\ <\ reserved\ word\ >$" clause, which provides the expression with additional semantics. In particular, *GEO* describes the spatial search geometry; *T_ICON_PATH* customizes the icon resource for the spatial search geometry; *T_LABEL* provides the icon label to be shown on the map; and *T_LABEL_SIZE* gives the size of

label in pixels. The corresponding spatial query results for the MapQL statement in Figure 1 is presented in Figure 2.

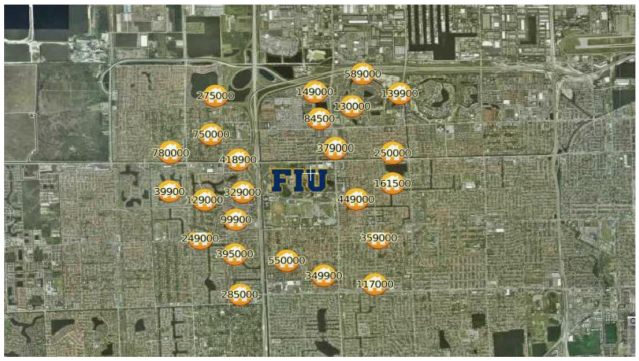


Fig. 2: The MapQL query result on real property data is displayed on the map.

Comparing with using GIS application programming interface (API), MapQL provides a better interface to facilitate the use of TerraFly map for both developers and end users without any functionality limitation. Similar to GIS API, MapQL enables users to flexibly create their own maps. However, our further study of TerraFly GeoCloud reveals three interesting and crucial issues which present similar challenges in other online spatial analysis systems.

The first issue is the difficulty in authoring MapQL queries. Though most of developers who are familiar with SQL can pick up MapQL quickly, the learning curve for end users who have no idea about SQL before is very steep. Authoring MapQL queries remains a challenges for the vast majority of users. As a result, it is difficult for those end users to utilize MapQL to complete a spatial analysis task from scratch.

The second issue is the complexity of a spatial analysis task. A typical spatial analysis task tends to involve a few sub-tasks. Moreover, those sub-tasks are not completely independent from each other, where the outputs of some sub-tasks are used as the inputs for other sub-tasks. According to the dependencies, a spatial data analysis task can be naturally presented as a workflow. The complexity of building such a workflow turns out to be a great obstacle for the users during the online spatial data analysis.

The third issue is the inefficiency of sequentially executing the workflow of a spatial analysis task. Even though the sub-tasks in a workflow are not linearly dependent on each other, the sub-tasks can only be sequentially executed by end users one by one. As a consequence, it fails to take advantage of the distributed environment to optimize the execution of independent sub-tasks in parallel.

The above three issues pose big challenges for users to freely and flexibly explore spatial data using online spatial analysis system. In this paper, we employ sequential pattern mining algorithms to discover the sequential query patterns from the MapQL query logs of TerraFly GeoCloud. With the help of discovered sequential query patterns, the workflows of spatial data analysis tasks are first constructed. FIU-Miner [13] is then employed to optimize the execution of the spatial data analysis tasks by maximizing the parallelization of sub-tasks in the corresponding workflow.

The rest of the paper is organized as follows. Section II presents the related work. Section III gives the overview of our system. Section IV introduces the details in mining sequential query patterns among MapQL query logs, the generation of query templates for MapQL statements, and the workflow construction for the spatial data analysis tasks. Section V presents our empirical study to show the efficiency and the effectiveness of our system. Finally Section VI concludes and discusses future works.

II. Related Work

Sequential pattern mining has been studied for decades. Several famous algorithms have been proposed to discover the sequential pattern in sequence data including apriori-based GSP [11], SPIRIT [4], Spade [12], FreeSpan [5], and PrefixSpan [8]. This paper adapts the PrefixSpan algorithm to find the frequent sequential pattern.

Mining patterns in SQL logs has received a lot of attention in recent years, especially with the advent of Big data era. The research efforts in [3], [1] employ the discovered patterns from the SQL logs to facilitate the SQL auto-completion with recommending snippets of query. However, almost all the reported studies only focus on the patterns within a single statement. In this paper, we focus on the sequential query pattern which consists of a sequence of statements.

TerraFly [10], [9] is a platform which supports query and visualization of geo-spatial data. This platform provides users with customized aerial photography, satellite imagery and various overlays, such as street names, roads, restaurants, services and demographic data. TerraFly API allows application developer to create various GIS applications, such as geospatial querying interface, map display with user-specific granularity, real-time data suppliers, demographic analysis, annotation, and route dissemination via autopilots. TerraFly GeoCloud [7] is a system built on the TerraFly system. One of important features of TerraFly GeoCloud is that it provides MapQL to support more flexible and complicated spatial data analysis. However, it requires end users to compose the MapQL statements.

FIU-Miner is a data mining platform which supports data analyst with a user-friendly interface to perform rapid data mining task configuration. Users can assemble the existing algorithms into a workflow without writing a single line of code. The platform also supports data mining in distributed and heterogeneous environments [13]. In this paper, we take advantage of FIU-Miner to optimize the execution of spatial data analysis.

III. The System Overview

To address the highlighted issues of TerraFly GeoCloudin Section I, the online spatial analysis system is optimized by integrating FIU-Miner framework, which is capable of assembling sub-tasks into a workflow in accordance with the dependencies of sub-tasks and scheduling each sub-task for

execution in distributed environment. The overview of the integrated system is given in Figure 3.

The system consists of four tiers: User Interface, Geo-Spatial Web Service, Computing Service and Storage.

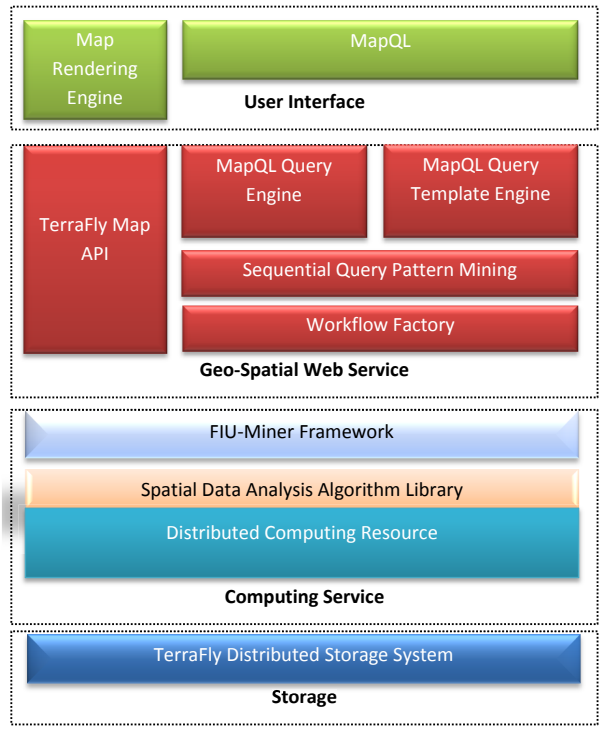


Fig. 3: The system overview.

In the layer of User Interface, Map Rendering Engine is responsible for rendering the geo-objects on the map nicely based on the visualization customized by users. The component of MapQL accepts MapQL statements that describe the spatial analysis task and the required elements for map rendering.

The second layer is Geo-Spatial Web Service. In this layer, TerraFly Map API provides the interface to access the spatial data for other components in the same layer and Map Rendering Engine in User Interface layer. MapQL Query Engine is responsible for analyzing the MapQL statements and guarantees their syntactic and semantic correctness. Sequential Query Pattern Mining is utilized to discover the sequential query pattern from the MapQL query log data. The discovered sequential query pattern can be used to generate the query templates by MapQL Query Template Engine. Users are able to rewrite the MapQL query template to construct new MapQL statements in User Interface layer. A sequential query pattern contains a sequence of MapQL queries and is used to form a workflow by Workflow Factory. Each query in a sequential pattern corresponds to a sub-task in the corresponding workflow.

The third layer is Computing Service. FIU-Miner Framework takes a workflow from the second layer as an input. FIU-Miner takes the load balance of distributed environment into account to schedule the sub-tasks of a workflow for execution. The spatial data analysis library is deployed in the distributed environment. The library can be extended by developers. The computing resource is used to support the spatial data analysis tasks.

The last layer is mainly responsible for storing and managing the spatial data. All the spatial data in TerraFly is stored in the distributed file system, where replica of data guarantees the safety and reliability of system.

In the subsequent sections, we introduce the detail of the proposed system.

IV. Sequential Query Pattern

In our system, users mainly use MapQL statements to accomplish their online spatial data analysis tasks. Although MapQL is powerful and flexible to satisfy the analysis requirement of the users, it requires end users to compose the statements, typically from scratch. Based on the user query logs, the sequential MapQL query pattern is proposed to partially address the problem.

A. Sequential MapQL Query Pattern

Let D be a collection of sequences of queries, denoted as $D = \{S_1, S_2, ..., S_n\}$, where S_i is a sequence of queries occurring within a session, ordered according to their time stamps. Therefore, $S_i =< q_1, q_2, ..., q_i, ..., q_m >$ is a sequence including m queries in temporal order. If q_i is a compound query composed of two sub-queries q_{i0} and q_{i1}, then $S_i =< q_1, q_2, ..., (q_{i0}, q_{i1}), ..., q_m >$. Sub-queries in a parenthesis are from a compound query occurring at the same time stamp.

A k-subsequence of S_i is a sequence of queries with length k denoted as $T =< t_1, t_2, ..., t_k >$, where each $t \in T$ corresponds to only one query $q \in S_i$, and all the queries in T are kept in temporal order. $T \sqsubseteq S_i$ is used to indicate that T is a subsequence of S_i.

Given the query sequence data collection D, a sequential query pattern is a query sequence whose occurrence frequency in the query log D is no less than a user-specified threshold $min\_support$. Formally, the support of sequence T is defined as

$$support(T) = |\{S_i | S_i \in D \wedge T \sqsubseteq S_i\}|.$$

A sequence T is a sequential query pattern, only if $support(T) \geq min\_support$.

The process of discovering all the sequential query patterns from the MapQL logs generally consists of two stages. The first stage is to generalize the representation of MapQL statements by parsing the MapQL text into syntax units. Based on the syntax representation of MapQL statements, the second stage is to mine the sequential query patterns from the sequences of MapQL statements.

B. Representation Of MapQL

As shown in Figure 3, MapQL Query Engine collects the MapQL statements and records them in the log files. A snippet of MapQL logs are given in Table I. Each MapQL statement is associated with a user session ID and a time stamp. All

the statements are organized in temporal order. Those MapQL statements sharing the same session ID are those issued by a user within a session. Our goal is to discover interesting patterns from the query logs. For example, according to the log data in the TABLE I, an interesting pattern is that users who viewed a particular street are more likely to look for the hotels along that street.

TABLE I: A snippet of MapQL logs.

Session ID	Timestamp	MapQL statement
1	20140301 13:26:33	SELECT geo FROM street WHERE name LIKE 'sw 8th';
1	20140301 13:28:26	SELECT h.name FROM street s LEFT JOIN hotel h ON $ST\_Distance(s.geo, h.geo) < 0.05$ WHERE s.name = 'sw8 th' AND h.star >= 4;
2	20140315 14:21:03	select geo from street where name like 'turnpike';
2	20140315 14:25:21	SELECT h.name FROM street s LEFT JOIN hotel h ON $ST\_Distance(s.geo, h.geo) < 0.05$ WHERE s.name = 'turnpike' AND h.star >= 4;
3	20140316 10:23:08	SELECT zip FROM us_zip;
4	20140319 11:19:21	SELECT count(*) FROM hotel;
...	...	...

In order to discover patterns from the query logs, intuitively, existing sequential pattern mining algorithms can be directly applied to the raw logs of MapQL statements, where different text of MapQL statements are treated as different items. However, representing an query item by the text of the MapQL statement is often too specific. As a consequence, it is difficult to discover the meaningful sequential patterns with such representations. For instance, the first and third records in TABLE I are identified as different query items during sequential query pattern mining, although both MapQL statements share the same semantics (i.e., locating a street given its partial name).

To address the aforementioned problem, the representation of a query in our system is generalized by parsing a MapQL statement into a collection of syntax units. The syntax units are organized as a syntax tree. For instance, the syntax tree for the first record of TABLE I is presented in Figure 4. There are two types of labels in the node of syntax tree. One is the type of a syntax unit, such as "Select Clause". The other label in the parenthesis is the content of a syntax unit, for example, "sw 8th". Provided with the syntax tree, the MapQL query can be generalized by representing any nodes with their types instead of their actual contents. For instance, assuming the node with 'Value' type in the syntax tree is represented as "#Value#" rather than using its text content, the original MapQL statements in both the first and third row of TABLE I are rewritten as "SELECT geo FROM street WHERE name LIKE #Value#;". Therefore, the two MapQL statements with the same semantics share the same query item. In addition, to simplify the extraction of patterns, each query item is identified with a unique integer number.

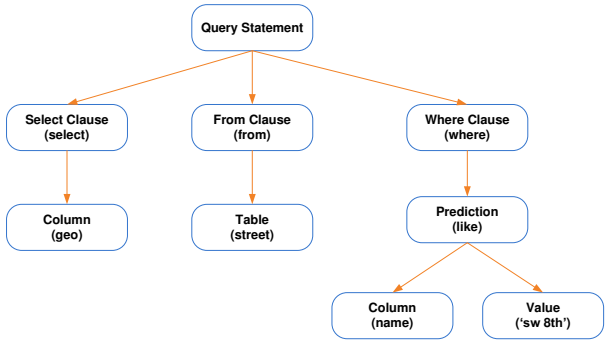


Fig. 4: The syntax tree for a MapQL statement " SELECT geo FROM street WHERE name LIKE 'sw 8th';".

C. Mining Sequential Query Pattern

Based on the properly generalized representation of a MapQL query, the PrefixSpan algorithm [8] is applied to efficiently discover all the sequential query patterns from the MapQL query log data.

The main idea of the PrefixSpan algorithm is to recursively partition the whole dataset into some sub-datasets, where the query sequences in a sub-dataset share the same prefix subsequence. The number of query sequences in each sub-dataset indicates the $support$ of its corresponding prefix subsequence. If a prefix subsequence T whose $support$ is no less than the user specified threshold $min\_support$, T is a sequential query pattern. Given two sequences T and R, $T \sqsubseteq R$ if T is a subsequence of R. An important property (i.e., downward closure property) is that R cannot be a sequential query pattern if T is not a sequential query pattern. According to the property, the recursive partition to search for super-pattern is not terminated until the size of current sub-dataset is smaller than the $min\_support$.

The PrefixSpan algorithm is illustrated in Figure 5. The top table presents the original collection of query sequences which contains two sequences of queries $S_1 =< (q_0, q_1) >$ and $S_2 =< (q_0), (q_2) >$. Sequence S_1 has only one compound query composed of q_0 and q_1. The other sequence S_2 has two queries named q_0 and q_2. Let $min\_support = 2$. The procedure of mining sequential query patterns is described as follows.

1) Find the frequent subsequences with a single query: $< q_0 >, < q_1 >, < q_2 >$.
2) Take the subsequences found in the step 1 as the prefixes. Based on the prefixes, the original dataset is partitioned into three sub-datasets, where each of them is specified by a prefix subsequence. The support of the prefix subsequence is the number of postfix sequences in its corresponding sub-dataset. The prefix patterns are extracted if their supports are larger than $min\_support$. Only the prefix subsequence in $D1$ is a sequential query pattern.

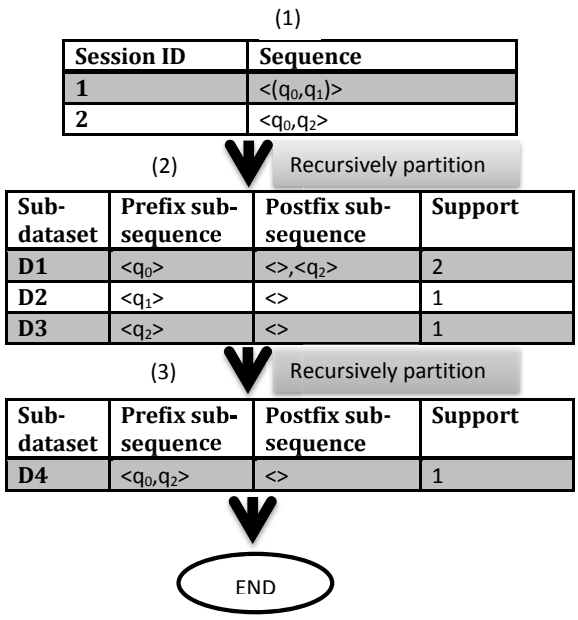


Fig. 5: An example illustrating the PrefixSpan algorithm.

3) Recursively partition $D1$. As a result, only one sub-dataset is generated and its support is 1.
4) Terminate the partition since no new prefix patterns can be further derived.

In the end, PrefixSpan discovers one sequential query pattern $< q_0 >$.

D. Query Template

Query template is generated by MapQL Query Template Engine in the system. This function alleviates the burden of users since MapQL queries can be composed by rewriting query templates. Based on the discovered sequential query patterns, a query template is generated by Algorithm 1. This algorithm scans the syntax trees in the sequential query pattern and replaces the specific table, column and constant value with template parameters. The algorithm guarantees that the same table, column or constant value appearing at multiple places, even multiple queries of a sequence acquires the same template parameter. Users can easily convert the template to executable queries by assigning the template parameters with specific values.

Given a sequential query pattern that contains the two queries with session ID 1 in TABLE I, we can apply Algorithm 1 to generate the template for the sequential query pattern. The generated template is shown in Figure 6. This template owns three parameters (i.e., #arg1#, #arg2#, #arg3#). Provided with values of these parameters, the executable sequence of queries can be easily derived from the template.

E. Spatial Data Analysis Workflow

All the MapQL queries in a sequential pattern are organized in a workflow, where the template parameters indicate the data transmission between queries. A sequence of queries

Algorithm 1 templateGen

1: **procedure** $templateGen(\mathbf{S})$
▷**Input:** **S** is a sequential pattern which contains a sequence of MapQL queries.
▷**Output:** query template for sequential query pattern.
2: initialize an empty inverted index TAB for Table
3: initialize an empty inverted index COL for Column
4: initialize an empty inverted index VAL for Value
5: **for** each query q_i in S **do**
6: **while** each node e in the syntax tree of q_i **do**
7: **if** $e.label$ is not generalized **then**
8: CONTINUE
9: **end if**
10: **if** $e.type$ is TABLE **then**
11: add element to TAB for the table name
12: **end if**
13: **if** $e.type$ is COLUMN **then**
14: add element to COL for the column name
15: **end if**
16: **if** $e.type$ is VALUE **then**
17: add element to VAL for the constant value
18: **end if**
19: **end while**
20: **end for**
21: With TAB, COL, VAL, rewrite S by replacing a specific table, column and constant value with corresponding template name.
22: **return** the rewritten sequence as template.
23: **end procedure**

```
Template(#arg1#, #arg2#, #arg3#):

  SELECT     geo
  FROM       street
  WHERE
             name LIKE #arg1#;

  SELECT     h.name
  FROM       street s
  LEFT JOIN  hotel h
  ON         ST_Distance(s.geo,h.geo) < #arg2#
  WHERE      s.name LIKE #arg1#
  AND        h.star >= #arg3#;
```

Fig. 6: Example of a generated template.

constitutes a spatial data analysis task and atypical spatial data analysis task often involves a few sub-tasks. The dependencies among those sub-tasks make spatial data analysis very complicated. The complexity of spatial data analysis dictates the support of workflow. In our system, Workflow Factory is designed and implemented in support of executing a complex spatial data analysis task in a workflow. A workflow is represented as a directed and connected graph consisting of nodes (denoting the sub-tasks) and edges (describing the dependencies among the sub-tasks). Data transmission between dependent sub-tasks

are supported in our system.

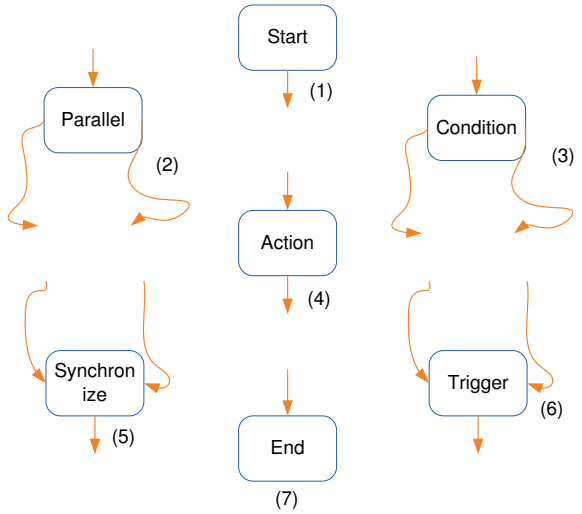


Fig. 7: Node types in a workflow.

In order to facilitate the spatial data analysis, we design seven types of nodes as shown in Figure7.

1) **Start Node** This type of node indicates the start of the workflow. There is only one such node in a valid workflow. This start node must link to one other node.
2) **Parallel Node** This type of node has one input link and more than one output links. After the work is completed in the parent node, the parallel node triggers the sub-tasks in its children nodes. All the sub-tasks of its children are executed in parallel.
3) **Condition Node** One input link and more than one output links are associated with this type of node. When the control flow reaches a condition node, it will check the input data and then move along one of its output links.
4) **Action Node** One input link and one output link are associated with this type of node. It often accommodates the sub-tasks for spatial data analysis. The data from the input link is fed into the sub-task and the result data of this sub-task is forwarded along its output link.
5) **Synchronize Node** This type of node has more than one input links and one output link. This node does not direct the control flow to its output link until all the sub-tasks in its parent nodes is completed.
6) **Trigger Node** More than one input links and one output link are associated with this type of node. The node starts the sub-tasks in its output link once one of sub-tasks in its parent nodes is finished.
7) **End Node** Any valid workflow should have one and only one end node. It indicates the end of the workflow.

Based on the generated template in Figure 6, two simple workflows can be constructed in Figure 8. These two workflows accomplish the same spatial data analysis task described in Figure 6. In the subfigure (1) of Figure 8, the two sub-tasks (i.e., SearchStreet and SearchHotel) are executed sequentially.

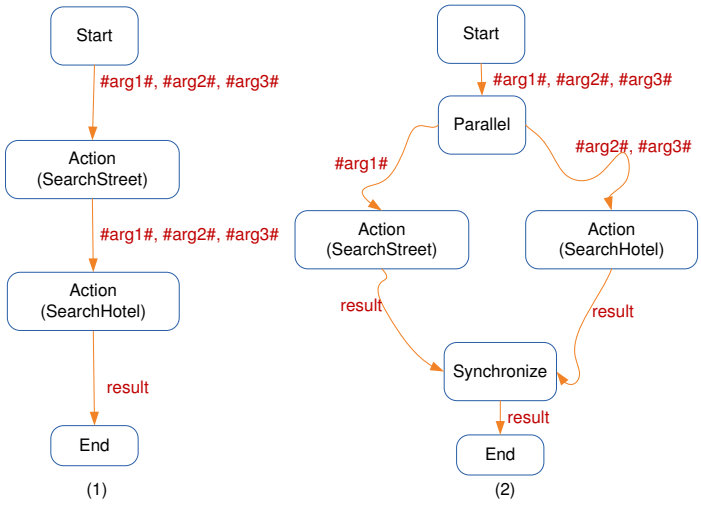


Fig. 8: Workflow examples.

However, SearchStreet needs the template parameter #arg1# as its input, while SearchHotel needs all three parameters. Provided with the three parameters, both sub-tasks can be executed independently. Thus, in the subfigure(2) of Figure 8, a parallel workflow is introduced to complete the spatial data analysis task. Since our data analysis tasks are scheduled by FIU-Miner, which takes full advantage of the distributed environment, the parallel workflow is more preferable to our system in terms of efficiency.

V. Empirical Study

In this section, the empirical study is conducted to demonstrate the efficiency and effectiveness of our system.

A. Setup

Besides providing the powerful API to support GIS applications, the TerraFly platform has a rich collection of GIS datasets. TerraFly owns the US and Canada roads data, the US Census demographic and socioeconomic data, the property lines and ownership data of 110 million parcels, 15 million records of businesses with company stats and management roles and contacts, 2 million physicians with expertise detail, various public place datasets, Wikipedia, extensive global environmental data, etc. Users can explore these datasets by issuing MapQL queries in our system. A case study on house property exploration is conducted to show how our system works.

B. House Property Exploration

The proposed system provides an optimized solution to spatial data analysis problem by explicitly constructing a workflow. It supports many different applications by analyzing the corresponding datasets. One typical application scenario is to locate the house property with a good appreciation potential for investment. Intuitively, it is believed that a property is deserved for investment if the price of the property is lower than the ones of surrounding properties. Our system is capable

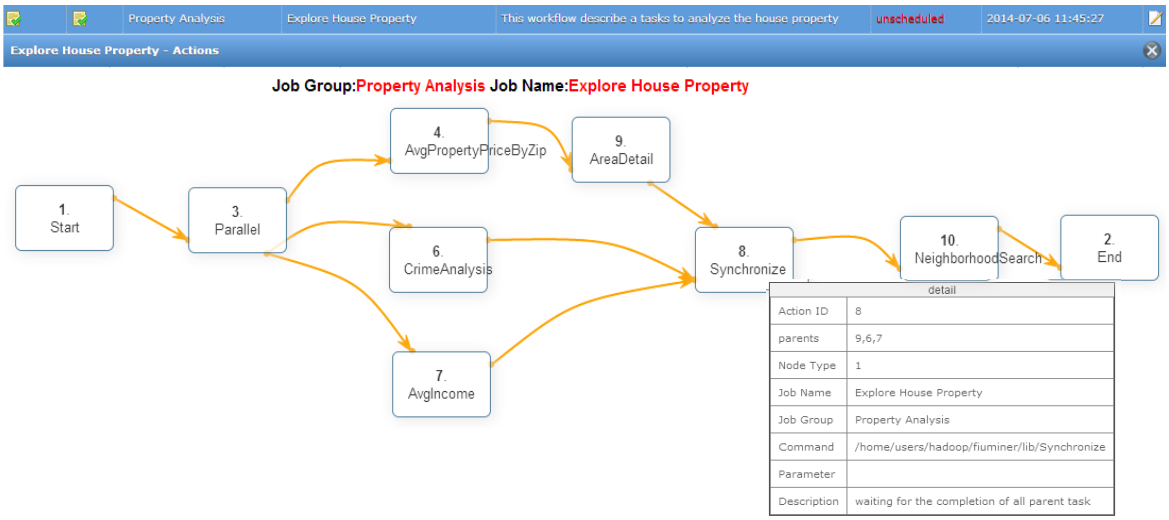


Fig. 9: The workflow of searching for house properties with a good appreciation potential. All the sub-tasks in the workflow are scheduled by FIU-Miner and are executed in the distributed environment.

of helping users (e.g.,investors) to easily and conveniently identify such properties. According to the historical query logs collected in our system, the sequential query patterns are extracted. Based on the discovered sequential query patterns, the query templates are then generated automatically. The templates related to the house property case study are assembled to build a workflow for house property data analysis. The workflow is presented in Figure9.

In the workflow, there are nine sub-tasks, denoted as rectangles, to constitute the complete house property analysis task. A user can view the detailed information of each sub-task from a pop-up layer as long as the mouse hovers on the corresponding node. The workflow begins with a start node, which is used to prepare the required setting and parameters. The start node links to the parallel node with three out links. The parallel node indicates that the three sub-tasks along its out links are able to be executed simultaneously.

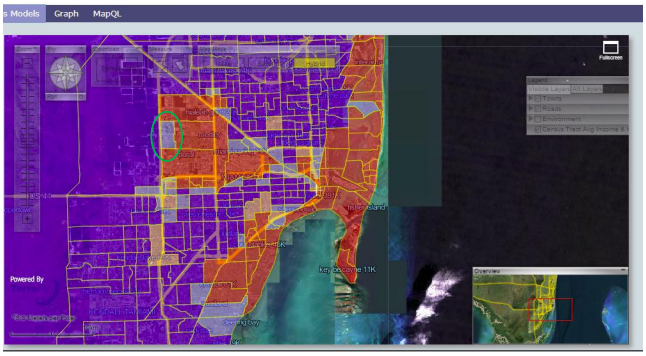


Fig. 10: Average property prices by zip code in Miami.

The AvgPropertyPriceByZip node in the workflow calculates the average property price. The overview of the analysis results is presented in Figure 10. Note that the property prices of red regions are higher than those of blue regions. From the overview, users often interested in the regions marked with a green circle since the average property price of the region is lower than the ones of its surroundings.

Fig. 11: Detailed properties in Miami.

In the next step, users check more detailed information on the region in the green circle by conducting the data analysis in the AreaDetail node. The spatial auto-correlation analysis on the average property prices by zip code data in Miami is conducted in this node and the analysis results are shown in Figure 11. Each point in the scatter plot corresponds to one zip code. Moran's I measure is applied during the auto-correlation analysis [2], [6]. The points in the first and third quadrants show positive associations with its surroundings, while the points in the second and fourth quadrants indicate negative associations. Herein, users are generally interested in the points of second quadrants, having lower property price than the ones of its surrounding regions. The interesting points are marked with yellow circles. The analysis leads to the result that most of the cheap properties with good appreciation potential are along the Gratigny Pkwy.

In order to make sure that the areas with cheap properties have good appreciation potential, spatial data analysis to investigate the crime rate and average income of these areas are conducted. The two data analysis sub-tasks are described in CrimeAnalysis and AvgIncome nodes, respectively. These two

sub-tasks are executed in parallel with the properties analysis. The Synchronize node waits for the completion of all three sub-tasks along the in links. Parallel execution accelerates the whole spatial data analysis and reduces the time cost.

```
SELECT
  CASE
    WHEN   h.pvalue >= 400000
      THEN '/var/www/cgi-bin/redhouse.png'
    WHEN   h.pvalue BETWEEN 200000 AND 400000
      THEN '/var/www/cgi-bin/bluehouse.png'
    WHEN   h.pvalue BETWEEN 100000 AND 200000
      THEN '/var/www/cgi-bin/bluehouse.png'
    ELSE   '/var/www/cgi-bin/darkhouse.png'
  END AS T_ICON_PATH,
  h.geo AS GEO
FROM
    osm_fl o
  LEFT JOIN
    south_florida_house_price h
  ON
    ST_Distance(o.geo, h.geo) < 0.05
WHERE
     o.name = #arg1# AND
     h.std_pvalue < 0 AND
     h.std_sl_pvalue > 0;
```

Fig. 12: A template for searching the neighborhood, given the partial name of street.

Without discovering any abnormalities in the crime rate and average income, users proceed to acquire more detailed property information along the Gratigny Pkwy by executing the sub-task in the NeighborhoodSearch node. The MapQL query listed in Figure 12 is executed in the NeighborhoodSearch node by passing the 'Gratigny Pkwy' as the input parameter. The MapQL statement employs different colors to mark the regions with various property prices. The final analysis results are presented in Figure 13. The regions painted in dark have the cheapest property prices and good appreciation potential.

Fig. 13: The final analysis results.

With the completion of the sub-task in the NeighborhoodSearch node, the control flow reaches the end node of the workflow. Comparing to the analysis procedure without workflow, where sub-tasks can only be executed sequentially, our system takes full advantage of FIU-Miner to schedule multiple tasks simultaneously in the distributed environments. It greatly reduces the time consumed by a complex spatial data analysis task and increases the throughput of our system.

VI. Conclusion

This paper proposes an approach to optimize spatial data analysis by integrating the FIU-Miner framework and the TerraFly system. Our approach makes use of sequential query patterns, which are discovered from the query logs, to facilitate the data analysis with query templates and optimized workflows. A case study is presented to demonstrate the effectiveness and efficiency of our system.

There are several future research directions to improve our current system such as developing more efficient and effective algorithms for discovering the query patterns and designing better techniques for generating query templates and constructing workflows.

Acknowledgment

The work is partially supported by the National Science Foundation under grants CNS-0821345, HRD-0833093, IIP-0829576, CNS-0959985, CNS-1126619, IIP-1338922, IIP-1237818, IIP-1330943, and IIS-1213026, the U.S. Department of Homeland Security under Award Number 2010-ST-062000039, the U.S. Department of Homeland Securitys VACCINE Center under Award Number 2009-ST-061-CI0001, and Army Research Office under grant number W911NF-1010366 and W911NF-12-1-0431.

References

[1] Javad Akbarnejad, Gloria Chatzopoulou, Magdalini Eirinaki, Suju Koshy, Sarika Mittal, Duc On, Neoklis Polyzotis, and Jothi S Vindhiya Varman. Sql querie recommendations. *VLDB*, 3(1-2):1597–1600, 2010.

[2] Luc Anselin. Local indicators of spatial associationłlisa. *Geographical analysis*, 27(2):93–115, 1995.

[3] Gloria Chatzopoulou, Magdalini Eirinaki, and Neoklis Polyzotis. Query recommendations for interactive database exploration. In *Scientific and Statistical Database Management*, pages 3–18, 2009.

[4] Minos N Garofalakis, Rajeev Rastogi, and Kyuseok Shim. Spirit: Sequential pattern mining with regular expression constraints. In *VLDB*, volume 99, pages 7–10, 1999.

[5] Jiawei Han, Jian Pei, Behzad Mortazavi-Asl, Qiming Chen, Umeshwar Dayal, and Mei-Chun Hsu. Freespan: frequent pattern-projected sequential pattern mining. In *ACM SIGKDD*, pages 355–359, 2000.

[6] Hongfei Li, Catherine A Calder, and Noel Cressie. Beyond moran's i: testing for spatial dependence based on the spatial autoregressive model. *Geographical Analysis*, 39(4):357–375, 2007.

[7] Yun Lu, Mingjin Zhang, Tao Li, Yudong Guang, and Naphtali Rishe. Online spatial data analysis and visualization system. In *SIGKDD Workshop on Interactive Data Exploration and Analytics*, pages 71–78, 2013.

[8] Jian Pei, Helen Pinto, Qiming Chen, Jiawei Han, Behzad Mortazavi-Asl, Umeshwar Dayal, and Mei-Chun Hsu. Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In *ICDE*, pages 215–224, 2001.

[9] N Rishe, M Gutierrez, A Selivonenko, and S Graham. Terrafly: A tool for visualizing and dispensing geospatial data. *Imaging Notes*, 20(2):22–23, 2005.

[10] Naphtali Rishe, Shu-Ching Chen, Nagarajan Prabakar, Mark Allen Weiss, Wei Sun, Andriy Selivonenko, and D Davis-Chu. Terrafly: A high-performance web-based digital library system for spatial data access. In *ICDE*, pages 17–19, 2001.

[11] Ramakrishnan Srikant and Rakesh Agrawal. *Mining sequential patterns: Generalizations and performance improvements*. Springer, 1996.

[12] Mohammed J Zaki. Spade: An efficient algorithm for mining frequent sequences. *Machine learning*, 42(1-2):31–60, 2001.

[13] Chunqiu Zeng, Yexi Jiang, Li Zheng, Jingxuan Li, Lei Li, Hongtai Li, Chao Shen, Wubai Zhou, Tao Li, Bing Duan. Fiu-miner: a fast, integrated, and user-friendly system for data mining in distributed environment. In *ACM SIGKDD*, pages 1506–1509, 2013.

Real time contextual collective anomaly detection over multiple data streams

Yexi Jiang, Chunqiu Zeng
School of Computing and Information Sciences
Florida International University
Miami, FL, USA
{yjian004, czeng001}@cs.fiu.edu

Jian Xu
School of Computer Science Technology and Engineering
Nanjing University of Science and Technology
Nanjing, China
dolphin.xu@njust.edu.cn

Tao Li
School of Computing and Information Sciences
Florida International University
Miami, FL, USA
taoli@cs.fiu.edu

ABSTRACT

Anomaly detection has always been a critical and challenging problem in many application areas such as industry, healthcare, environment and finance. This problem becomes more difficult in the Big Data era as the data scale increases dramatically and the type of anomalies gets more complicated. In time sensitive applications like real time monitoring, data are often fed in streams and anomalies are required to be identified online across multiple streams with a short time delay. The new data characteristics and analysis requirements make existing solutions no longer suitable.

In this paper, we propose a framework to discover a new type of anomaly called contextual collective anomaly over a collection of data streams in real time. A primary advantage of this solution is that it can be seamlessly integrated with real time monitoring systems to timely and accurately identify the anomalies. Also, the proposed framework is designed in a way with a low computational intensity, and is able to handle large scale data streams. To demonstrate the effectiveness and efficiency of our framework, we empirically validate it on two real world applications.

1. INTRODUCTION

Anomaly detection is one of the most important tasks in data-intensive applications such as the healthcare monitoring [22], stock analysis [26], disaster management [32], system anomaly detection [24], and manufacture RFID management [4], etc. In the Big Data era, the aforementioned applications often require real-time processing. However, existing data processing infrastructures are designed based on inherent non-stream programing paradigm such as MapReduce [11], Bulk Synchronous Parallel (BSP) [30], and their variations. To reduce the processing delay, these applications have gradually migrated to stream processing engines [27, 10, 1, 9]. As the infrastructures have been changed, anomalies in these applications are required to be identified online across multiple data streams. The new data characteristics and analysis requirements make existing anomaly detection solutions no longer suitable.

ODD'14, August 24 - 27 2014, New York, NY, USA.

http://dx.doi.org/10.1145/2656269.2656271.

1.1 A Motivating Example

EXAMPLE 1. *Figure 1 illustrates the scenario of monitoring a 6-node computer cluster, where the x-axis denotes the time and the y-axis denotes the CPU utilization. The cluster has been monitored during time* $[0, t_6]$. *At time* t_2, *a computing task has been submitted to the cluster and the cluster finishes this task at time* t_4. *As shown, two nodes (marked in dashed line) behave differently from the majority during some specific time periods. Node ① has a high CPU utilization during* $[t_1, t_2]$ *and a low CPU utilization during* $[t_3, t_4]$ *while node ② has a medium CPU utilization all the time. These two nodes with their associated abnormal periods are regarded as anomalies. Besides these two obvious anomalies, there are a slight delay on node ③ due to the network delay and a transient fluctuation on node ④ due to some random factors. However, they are normal phenomena in distributed systems and are not regarded as anomalies.*

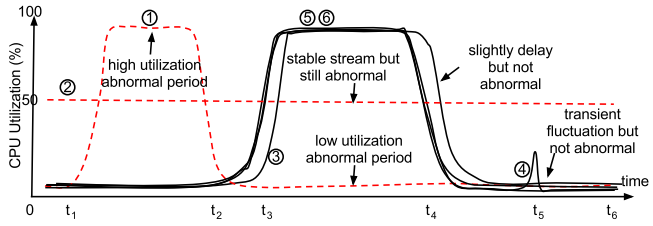


Figure 1: CPU utilization of a computing cluster

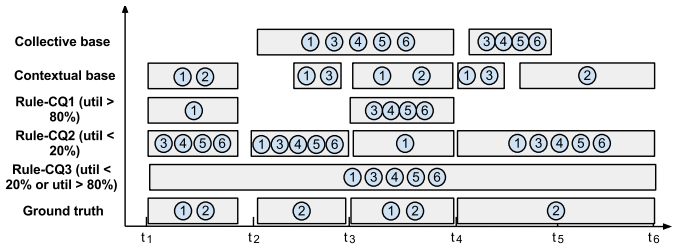


Figure 2: Identified anomalies in Example 1 (The box lists the IDs of abnormal streams during specified time period)

A quick solution for stream based anomaly detection is to leverage the techniques of *complex event processing (CEP)* [23, 2] by expressing the anomalies detection rules with corresponding continuous query statements. This rule-based detection method can be applied to the scenarios where the anomaly can be clearly defined. Besides using CEP, several stream based anomaly detection algorithms have also been proposed. They either focus on identifying contextual anomaly over a collection of stable streams [7] or collective anomaly from one stream [3, 25]. These existing methods are useful in many applications but they still cannot identify certain types of anomalies. A simple example of such scenario is illustrated in Example 1.

Figure 2 plots the ground truth as well as all the anomalies identified by existing methods including the CEP query with three different rules (Rule-CQ1, 2, and 3), the collective based anomaly detection [6], and contextual based anomaly detection [8].

To detect the anomalies via CEP query, the idea is to capture the events when the CPU utilizations of nodes are too high or too low. An example query following the syntax of [2] can be written as follows:

```
PATTERN SEQ(Observation o[])
WHERE avg(o[].cpu) oper threshold
  (AND|OR avg(o[].cpu) oper threshold)*
WITHIN {length of sliding window}
```

where the selection condition in `WHERE` clause is the conjunction of one or more boolean expressions, *oper* is one of {>, <, <>, ==}, and `threshold` can be replaced by any valid expression. However, CEP queries are unable to correctly identify the anomalies in Figure 1 no matter how the selection conditions are specified. For instance, setting the condition as `avg(o[].cpu) > {threshold}` would miss the anomalies during $[t_3, t_4]$ (Rule-CQ1); setting the condition as `avg(o[].cpu) < {threshold}` would miss the anomalies during $[t_1, t_2]$ (Rule-CQ2); and combining the two above expressions with OR still does not work (Rule-CQ3). Besides deciding the selection condition, how to rule out the situations of slight delays and transient fluctuations, and how to set the length of the sliding windows are all difficult problems when writing the continuous queries. The main reason is that the continuous query statement is not suitable to capture the contextual information where the "normal" behaviors are also dynamic (the utilizations of normal nodes also change over time in Figure 1).

Compared with CEP based methods, contextual anomaly detection methods (such as [14, 17]) achieve a better accuracy as they utilize the contextual information of all the streams. However, one limitation of contextual based methods is that they do not leverage the temporal information of streams and are not suitable for anomaly detection in dynamic environments. Therefore, these methods would wrongly identify the slightly delayed and fluctuated nodes as anomalies.

For the given example, collective anomaly detection methods do not work well neither. This is because these methods would identify the anomaly of each stream based on its normal behaviors. Once the current behavior of a stream is different from its normal behaviors (identified based on historical data), it is considered as abnormal. In the example, when the cluster works on the task during $[t_3, t_4]$, all the working nodes would be identified as abnormal due to the sudden burst.

1.2 Contributions

In this paper, we propose an efficient solution to identify this special type of anomaly in Example 1, named *contextual collective anomaly*. Contextual collective anomalies bear the characteristics of both contextual anomalies and collective anomalies. This type of anomaly is common in many applications such as system monitoring, environmental monitoring, and healthcare monitoring, where data come from distributed but homogeneous data sources. We will formally define this type of anomaly in Section 2.

Besides proposing an algorithm to discover the contextual collective anomalies over a collection of data streams, we also consider the scale-out ability of our solution and develop a distributed streaming processing framework for contextual collective anomaly detection. More concretely, our contributions can be described as follows:

- We provide the definition of contextual collective anomaly and propose an incremental algorithm to discover the contextual collective anomalies in real time. The proposed algorithm combines the contextual as well as the historical information to effectively identify the anomalies.
- We propose a flexible three-stage framework to discover such anomalies from multiple data streams. This framework is designed to be distributed and can be used to handle large scale data by scaling out the computing resources. Moreover, each component in the framework is pluggable and can be replaced if a better solution is proposed in the future.
- We empirically demonstrate the effectiveness and efficiency of our solution through the real world scenario experiments.

The rest of the paper is organized as follows. Section 2 gives a definition of contextual collective anomaly and then presents the problem statement. Section 3 provides an overview of our proposed anomaly detection framework. We introduce the three-stage anomaly detection algorithm in detail in Section 4. Section 5 presents the result of experimental evaluation. The related works are discussed in Section 6. Finally, we conclude in Section 7.

2. PROBLEM STATEMENT

In this section, we first give the notations and definitions that are relevant to the anomaly detection problem. Then, we formally define the problem based on the given notations and definitions.

DEFINITION 1. DATA STREAM. *A data stream S_i is an ordered infinite sequence of data instances $\{s_{i1}, s_{i2}, s_{i3}, ...\}$. Each data instance s_{it} is the observation of data stream S_i at timestamp t.*

The data instances s_{it} in S_i can have any number of dimensions, depending on the concrete applications. For the remaining of this paper, the terms "data instance" and "observation" would be used interchangeably. *To make the notation uncluttered, we use s_i in places where the absence of timestamp does not cause the ambiguity.*

DEFINITION 2. STREAM COLLECTION. *A stream collection $\mathcal{S} = \{S_1, S_2, ..., S_n\}$ is a collection of data streams. The number of streams $|\mathcal{S}| = n$.*

The input of our anomaly detection framework is a *stream collection*. For instance, in example 1, the input stream collection is $\mathcal{S} = \{①, ②, ③, ④, ⑤, ⑥\}$

DEFINITION 3. SNAPSHOT. *A snapshot is a set of key-value pairs $S^{(t)} = \{S_i : s_{it} | S_i \in \mathcal{S}\}$, denoting the set of the observations $\{s_{1t}, s_{2t}, \cdots, s_{|S|t}\}$ of the data streams in stream collection $\mathcal{S}$ at time t.*

A *snapshot* captures the configuration of each data stream in the stream collection for a certain moment. Taking Figure 1 for example, the snapshot at time t_5 is $S^{(t_5)} = \{① : 0\%, ② : 50\%, ③ : 0\%, ④ : 20\%, ⑤ : 0\%, ⑥ : 0\%\}$. For simplicity, we use $S(i)$ to denote the ith dimension of the observations in a certain snapshot. Note that all the observations in a snapshot have the same dimension.

DEFINITION 4. CONTEXTUAL COLLECTIVE ANOMALY. *A contextual collective stream anomaly is denoted as a tuple $< S_i, [t_b, t_e], N >$, where S_i denote a data stream from the collection of data streams $\mathcal{S}$, $[t_b, t_e]$ is the associated time period when S_i is observed to constantly deviate from the majority streams in S, and N indicates the severity of the anomaly.*

In Example 1, 3 contextual collective anomalies can be found in total. During time period $[t_1, t_2]$, node ① behaves constantly different from the other nodes, so there is an anomaly $<①, [t_1, t_2], N_1>$. The other two contextual collective anomalies, $<①, [t_3, t_4], N_2>$ and $<②, [0, t_6], N_3>$, can also be found with the same reason.

In Definition 4, the severity of deviation is measured in a given metric space $\mathcal{M}$ with a distance function $f : s_{it} \times s_{ku} \to \mathbb{R}$. For simplicity, we use Euclidean distance as an example throughout this paper.

Problem Definition. The anomaly detection problem in our paper can be described below: Given a stream collection $\mathcal{S} = \{S_1, S_2, ..., S_m\}$, identify the source of the contextual collective anomalies S_i, the associated time period $[t_s, t_e]$, as well as a quantification about the confidence of the detection p. Moreover, the detection has to be conducted on data streams that look-back is not allowed and the anomalies are able to be identified in real time.

3. FRAMEWORK OVERVIEW

In this section, we briefly describe how the aforementioned problem is addressed and then introduce the proposed distributed real time anomaly detection framework from a high level perspective.

As previously mentioned, *this paper focuses on discovering contextual collective anomalies over a collection of data streams obtained from a homogeneous distributed environment*. An example of the homogeneous distributed environment is the system with load balance, which is widely used at the backend by the popular web sites like Google, Facebook, and Amazon, etc.

It is known that, in such a kind of environment, the components should behave similar to each other. Therefore, the snapshots (the current observations) of these streams should be close to each other at any time. Naturally, we need to identify the anomalies by investigating both contextual information (the information of the current snapshot) and collective information (the historical information).

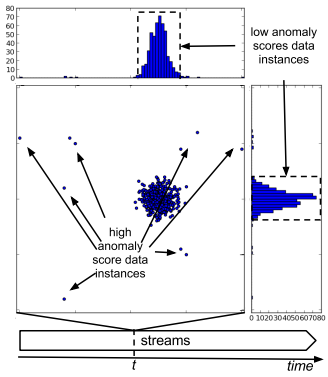


Figure 3: The snapshot at a certain timestamp

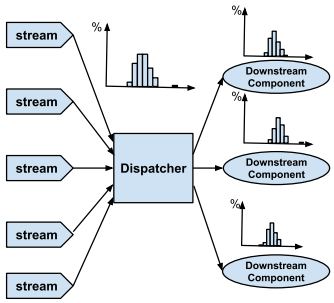


Figure 4: Random partition of data instance

The anomaly detection is conducted in three stages: the *dispatching stage*, the *scoring stage*, and the *alert stage*. The functionality of the three stages are briefly described as follows:

- *Dispatching stage:* This stage uses *dispatchers* to receive the observations from external data sources and then shuffle the observations to different downstream processing components.
- *Scoring stage:* This stage quantifies the candidate anomalies using *snapshot scorer* and then *stream scorer*.

  The *snapshot scorer* leverages contextual information to quantify the confidence of anomaly for each data instance at a given *snapshot*. Taking Figure 3 for example, it shows the data distribution by taking the snapshot of the 2-dimensional data instances of 500 streams at timestamp t. As shown, most of the data instances are close to each other and located in a dense area. These data instances are not likely to be identified as anomalies as their instance anomaly scores are small. On the contrary, a small portion of the data instances (those points that are far away from the dense region) have larger instance anomaly scores and are more likely to be abnormal.

  A data instance with a high anomaly score does not indisputably indicate its corresponding stream to be a real anomaly. This is because the transient fluctuation and phase shift are common in real world distributed environment. To mitigate such effects, the *stream scorer* is designed to handle the problem. In particular, the *stream scorer* combines the information obtained from the *instance scorer* and the historical information of each stream to quantify the anomaly confidence of each stream.
- *Alert stage:* The alert stage contains the *alter trigger*. The alert triggers leverage the unsupervised learning methods to identify and report the outliers.

The advantage of our framework is reflected by the ease of integration, the flexibility, and the algorithm independence.

Firstly, any external data sources can be easily fed to the framework for anomaly detection. Moreover, the components in every stage can be scaled-out to increase the processing capability if necessary. The number of components in each stage can be easily customized according to the data scale of concrete applications. Furthermore, the algorithms in each stage can be replaced and upgraded with better alternatives and the replacement would not interfere with other stages.

4. METHODOLOGY

4.1 Data Receiving and Dispatching

The dispatching stage is an auxiliary stage in our framework. When the data scale (i.e., the number of streams) is too large for a single computing component to process, the dispatcher would shuffle the received observations to downstream computing components in the scoring stage (as shown in Figure 4). By leveraging random shuffling algorithm like Fisher-Yates shuffle [12], dispatching can be conducted in constant time per observation. After dispatching, each downstream component would conduct scoring independently on a sampled stream observations with identical distribution.

Ideally, the observations coming from homogeneous data sources have very similar measurable value (e.g. workload of each server in a load balanced system), but random factors can easily cause the variations of the actual observations. Therefore in fact, an observation $\boldsymbol{s_i} \in \mathbb{R}^d$ is viewed as the ideal case value $\boldsymbol{s_{ideal}}$ with additive Gaussian noise so that $\boldsymbol{s_i} = \boldsymbol{s_{ideal}} + \boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. For those data sources in abnormal conditions, their observations are generated according to a different but unknown distributed. It is not difficult to know that, given enough observations, the mean and covariance can be easily estimated locally via maximum likelihood, i.e. $\hat{\boldsymbol{\mu}}_{ML} = \frac{\sum_i \boldsymbol{s_i}}{n}$ and $\widehat{cov}(S(x), S(y))_{ML} = \frac{1}{n-1}\sum_{i=1}^{n}(s_i(x) - \bar{S}(x))(s_i(y) - \bar{S}(y))$, where $\bar{S}(x)$ and $\bar{S}(y)$ respectively denote the sample mean of dimension x and y.

4.2 Snapshot Anomaly Quantification

Quantifying the anomaly in a snapshot is the first task in the *scoring stage* and we leverage *snapshot scorer* in this step. This score measures the amount of deviation of the specified observation s_{it} to the center of all the observations in a snapshot $S^{(t)}$ at timestamp t.

To quantify the seriousness of snapshot anomaly, we propose a simple yet efficient method. The basic idea is that the anomaly score of an observation is quantified as the amount of uncertainty it brings to the snapshot $S^{(t)}$. As the observations in a snapshot follows the normal distribution, it is suitable to use the increase of entropy to measure the anomaly of an observation. To quantify the anomaly score, two types of variance are needed: the *variance* and the *leave-one-out variance*, where the leave-one-out variance is the variance of the distribution when one specific data instance is not counted.

A naive algorithm to quantify the anomaly scores requires quadratic time ($O(dn + dn^2)$). By reusing the intermediate results, we propose an improved algorithm with time complexity linear to the number of streams. The pseudo code of the proposed algorithm is shown in Algorithm 1. As illustrated, matrix M is used to store the distances between each

Algorithm 1 Snapshot Anomaly Quantification

1. **INPUT**: Snapshot $S^{(t)} = (s_1, \cdots, s_n)$, where $s_i \in \mathbb{R}^d$.
2. **OUTPUT**: Snapshot anomaly scores $N \in \mathbb{R}^n$.
3. Create a $d \times n$ matrix $M = (s_1, \cdots, s_n)$
4. Conduct 0-1 normalization on rows of M.
5. $\boldsymbol{x} \leftarrow \mathbf{0}^d$ and $N = \mathbf{0}^d$
6. $\boldsymbol{m} \leftarrow (\mathbb{E}(S(1)), \cdots, \mathbb{E}(S(d))^T$
7. $M \leftarrow (s_{1t} - \boldsymbol{m}, s_{2t} - \boldsymbol{m}, \cdots, s_{dt} - \boldsymbol{m})$
8. **for** $j \leftarrow 0$ to d **do**
9. $\quad \boldsymbol{x}_j = ||j$th column of $M||_2^2$
10. **end for**
11. **for all** $s_i \in S^{(t)}$ **do**
12. $\quad$ Calculate N_i according to Equation (1).
13. **end for**
14. Conduct 0-1 normalization on N.
15. **return** N

dimension of the observations to the corresponding mean. Making use of M, the *leave-one-out variance* can be quickly calculated as $\sigma_{ik}^2 = \frac{n\sigma_k^2 - M_{i,k}^2}{n-1}$, where σ_k^2 denotes the variance of dimension k and σ_{ik}^2 denotes the leave-one-out variance of dimension k by excluding s_i. As the entropy of normal distribution is $H = \frac{1}{2}\ln(2\pi e\sigma^2)$, the increase of entropy for observation s_i at dimension k can be calculated as

$$d_k = H'_k - H_k = \ln\frac{\sigma_{ik}^2}{\sigma_k^2} = \ln\frac{(\boldsymbol{x}_k - M_{ik}^2)/(n-1)}{\boldsymbol{x}_j/n}, \quad (1)$$

where H'_k and H_k respectively denote the entropy of the snapshot distribution if s_i is not counted or counted.

Summing up all dimensions, the snapshot anomaly score of s_{it} is $N_{it} = \sum_k d_k$. Note that the computation implicitly ignores the correlation between dimensions. The reason is that if an observation is an outlier, the correlation effect would only deviate it further from other observations.

4.3 Stream Anomaly Quantification

As a stream is continuously evolving and its observations only reflect the transient behavior, snapshot anomaly score alone would result in a lot of false-positives due to the transient fluctuation and slight phase shift. To mitigate such situations, it is critical to quantify the stream anomaly by incorporating the historical information of the stream.

An intuitive way to solve this problem is to calculate the stream anomaly score from the recent historical instances stored in a sliding window. However, this solution has two obvious limitations: (1) It is hard to decide the window length. A long sliding window would miss the real anomaly while a short sliding window cannot rule out the false-positives. (2) It ignores the impact of observations that are not in the sliding window. The observation that is just popped out from the sliding window would immediately and totally lose its impact to the stream.

To well balance the history and the current observation, we use *stream anomaly score* N_i to quantify how significant a stream S_i behaves differently from the majority of the streams. To quantify N_i, we exploit the exponential decay function to control the influence depreciation. Supposing Δt is the time gap between two adjacent observations, *the influence of an observation* s_{it} *at timestamp* $t_{x+k} = t_x + k\Delta t$ can be expressed as $N_{it_x}(t_{x+k}) = N_{it_x}(t_x + k\Delta t) =$

$N_{it_x}e^{-\lambda kt}, (\lambda > 0)$, where λ is a parameter to control the decay speed. In the experiment evaluation, we will discuss how this parameter affects the anomaly detection results.

To make the notation uncluttered, we use t_{-i} to denote the timestamp that is $i\Delta t$ ahead of current timestamp t, i.e. $t_{-i} = t - i\Delta t$. Summing up the influences of all the historical observations, the overall historical influence I_{it} for current timestamp t can be expressed as Equation (2).

$$\begin{aligned} I_{it} &= N_{it_{-1}}(t) + N_{it_{-2}}(t) + N_{it_{-3}}(t) + ... \\ &= N_{it_{-1}}e^{-\lambda} + N_{it_{-2}}e^{-2\lambda} + N_{it_{-3}}e^{-3\lambda} + ... \\ &= e^{-\lambda}(N_{it_{-1}} + e^{-\lambda}(N_{it_{-2}} + e^{-\lambda}(N_{it_{-3}} + ... \\ &= e^{-\lambda}(N_{it_{-1}} + I_{it_{-1}}). \end{aligned} \tag{2}$$

The stream anomaly score of stream S_i is the summation of the data instance anomaly score of current observation N_{it} and the overall historical influence, i.e.,

$$N_i = N_{it} + I_{it}. \tag{3}$$

As shown in Equation (2), the overall historical influence can be incrementally updated with cost $O(1)$ for both time and space complexity. Therefore, *stream anomaly scorer* can be efficiently computed.

4.3.1 Properties of Stream Anomaly Score

The properties of stream anomaly score make our framework insensitive to the transient fluctuation and effective to capture the real anomaly.

Comparing to the transient fluctuation, the real anomaly is more durable. Figure 5 shows the situations of a transient fluctuation (in the left subfigure) and a real anomaly(in the right subfigure). In both situations, the stream behaves normally before timestamp t_x. For the left situation, a transient fluctuation occurs at timestamp t_{x+1}, and then the stream returns to normal at timestamp t_{x+2}. For the right situation, an anomaly begins at timestamp t_{x+1}, lasts for a while till timestamp t_{x+k}, and then the stream returns to normal afterwards. Based on Figure 5, we show two properties about the stream anomaly score.

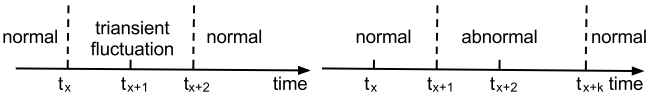


Figure 5: Transient Fluctuation and Anomaly

PROPERTY 1. *The increase of stream anomaly score caused by transient disturbance would decrease over time.*

PROPERTY 2. *The increase of stream anomaly score caused by anomaly would be accumulated over time.*

Similar properties can also be shown for the situation of slight shifts. A slight shift can be treated as two transient fluctuations occur at the beginning and the end of the shift. In the next section, we will leverage these two properties to effectively identify the anomalies in the ALERT STAGE.

4.4 Alert Triggering

Most of the stream anomaly detection solutions [13] identify the anomalies by picking the streams with top-k anomaly scores or the ones whose scores exceed a predefined threshold. However, these two approaches are not practical in real world applications for the following reasons: (1) *Threshold is hard to set.* It requires the users to understand the underlying mechanism of the application to correctly set the parameter. (2) *The number of anomalies are changing all the time.* It is possible that more than k anomaly streams exist at one time, then the top-k approach would miss these real anomalies.

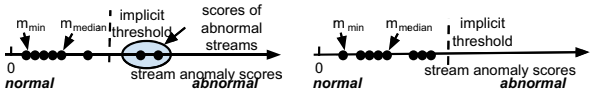


Figure 6: Abnormal Streams Identification

To eliminate the parameters, we propose an unsupervised method to identify and quantify the anomalies by leveraging the distribution of the anomaly scores. The first step is to find the median of the stream anomaly scores (N_{median}). If the distance between a stream anomaly score and the median score is larger than the distance between the median score and the minimal score (N_{min}), the corresponding stream is regarded as abnormal. As shown in Figure 6, this method implicitly defines a dynamic threshold (shown as the dashed line) based on the hypothesis that there is no anomaly. If there is no anomaly, the skewness of the anomaly score distribution should be small and the median score should be close to the mean score. If the hypothesis is true, $N_{median} - N_{min}$ should be close to half of the distance between the minimum score and the maximum score. On the contrary, if a score N_i is larger than $2 \times (N_{median} - N_{min})$, the hypothesis is violated and all the streams with scores at least N_i are abnormal.

Besides the general case, we also need to handle one special case: a transient fluctuation occurs at the current timestamp. According to Property (1) in Section 4.3.1, the effect of transient fluctuation is at most $d_{upper} = N_{it_{x+1}} - N_{jt_{x+1}}$ and it will monotonically decrease. Therefore, even a stream whose anomaly score is larger than $2 \times (N_{median} - N_{min})$, it can still be a normal stream if the difference between its anomaly score and N_{min} is smaller than d_{upper}. To prune the false-positive situations caused by transient fluctuation, the stream is instead identified as abnormal if

$$N_i > max(2(N_{median} - N_{min}), N_{min} + d_{upper}). \tag{4}$$

Another thing needs to be noted is that the stream anomaly scores have a upper bound $\frac{d_{upper}}{1-e^{-\lambda}}$. According to the property of convergent sequence, the stream anomaly scores of all streams would converge to this upper bound. When the values of stream anomaly scores are close to the upper bound, they tend to be close to each other and hard to be distinguished. To handle this problem, we reset all the stream anomaly scores to 0 whenever one of them close to the upper bound.

In terms of the time complexity, the abnormal streams can be found in $O(n)$ time. Algorithm 2 illustrates the algorithm of stream anomalies identification. The median of the scores can be found in $O(n)$ in the worst case using the *BFPRT algorithm [5]*. Besides finding the median, this algorithm also partially sorts the list by moving smaller scores before the median and larger scores after the median, making it trivial to identify the abnormal streams by only checking the streams appearing after the median.

5. EXPERIMENTAL EVALUATION

Algorithm 2 Stream Anomaly Identification

1. **INPUT**: λ, and unordered stream profile list $\mathcal{S} = \{S_1, ..., S_n\}$.
2. $mIdx \leftarrow \lceil \frac{|\mathcal{S}|}{2} \rceil$
3. $N_{median} \leftarrow \text{BFPRT}(\mathcal{S}, mIdx)$
4. $N_{min} \leftarrow \min(S_i.score | 0 \leq i \leq mIdx)$
5. $N_{max} \leftarrow N_{median}$
6. **for** $i \leftarrow mIdx$ to $|\mathcal{S}|$ **do**
7. **if** Condition (4) is satisfied **then**
8. Trigger alert for S_i with score N_i at current time.
9. **if** $N_i > N_{max}$ **then**
10. $N_{max} \leftarrow N_i$
11. **end if**
12. **end if**
13. **end for**
14. **if** N_{max} is close to the upper bound **then**
15. Reset all stream anomaly scores.
16. **end if**

To investigate the effectiveness and efficiency of our framework, we design several sets of experiments with one real world data applications: *anomaly detection over a computing cluster*. Taking these two applications as case studies, we show that our proposed framework can effectively identify the abnormal behavior of streams. It should be pointed out that our proposed framework can also be applied to many other application areas such as PM2.5 environment monitoring, healthcare monitoring, and stock market monitoring.

System anomaly detection is one of the critical tasks in system management. In this set of experiments, we show that our proposed framework can effectively and efficiently discover the abnormal behaviors of the computer nodes with high precision and low latency.

5.1 Experiment Settings

For the experiments, we leverage a distributed system monitoring tool [31] into a 16-node computing cluster. Then we deploy the proposed anomaly detection program on an external computer to analyze the collected trace data in real time. To well evaluate our proposed framework, we terminate all the irrelevant processes running on these nodes. On this node, we intentionally inject various types of anomalies and monitor their running status for 1000 seconds. The source code of injection program is available at `https://github.com/yxjiang/system-noiser`. The details of the injections are listed in Table 1.

Table 1: List of Injections

No.	Time Period	Node	Description
1	[100, 150]	2	Keep CPU utilization above 95.
2	[300, 400]	3	Keep memory usage at 70%.
3	[350, 400]	3	Keep CPU utilization above 95%.
4	[600, 650]	4	Keep memory usage at 70%.
5	[900, 950]	2,5	Keep CPU utilization above 95%.
6	[800, 850]	1-5,7-16	Keep CPU utilization above 95%.

Through these injections, we can answer the following questions about our framework: (1) Whether our framework can identify the anomalies with different types of root causes. (2) Whether our framework can identify multiple anomalies occurring simultaneously.

5.2 Results Analysis

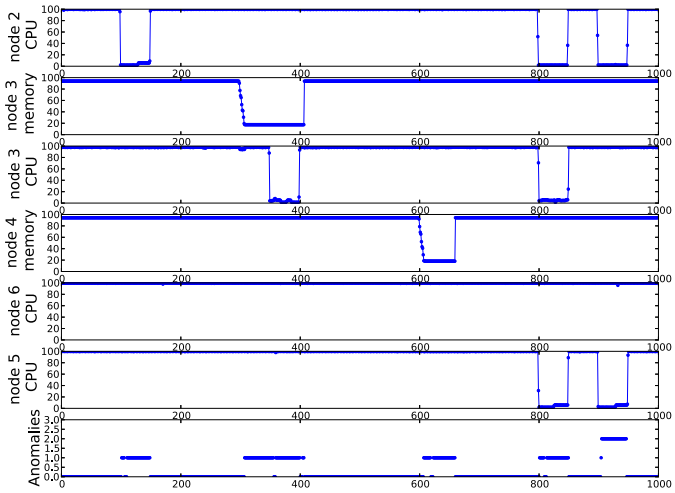


Figure 7: Injections and the captured alerts

Figure 7 illustrates the results of this experiment by plotting the actual injections (top 6 sub-figures) as well as the captured alerts (the bottom subplots), where the x-axis represents the time and y-axis represents the idled CPU utilization, idle memory usage or the number of anomalies in each timestamp. We evaluate the framework from 5 aspects through carefully-designed injections.

1. *Single dimension (e.g. idle CPU utilization or idle memory usage) of a single stream behaves abnormally.* This is the simplest type of anomalies. It is generated by injections No.1 and No.4 in Table 1. As shown in Figure 7, our framework effectively identifies these anomalies with the correct time periods.

2. *Multiple dimensions (e.g. CPU utilization and memory usage) of a single stream behaves abnormally at the same time.* This type of anomalies is generated by injections No.2 and No.3 in Table 1, and our framework correctly captures such anomalies during the time period [300, 400]. One thing should be noted is that the stream anomaly score of node 3 increases faster during the time period [350, 400] than the time period [300, 350]. This is because two types of anomalies (CPU utilization and memory usage) appear simultaneously during the time period [350, 400].

3. *Multiple streams behave abnormally simultaneously.* This type of anomalies is generated by injection No.5. During the injection time period, our framework correctly identifies both anomalies (on node 2 and node 5).

4. *Stable but abnormal streams.* This kind of anomaly is indirectly generated by injection No.6 in Table 1. This injection emulates the scenario that all the nodes but one (i.e., node 6) in a cluster received the command of executing a task. As is shown, although the CPU utilization of node 6 behaves stable all the time, it is still considered to be abnormal during the time period [800, 850]. This is because it remains idle when all the other nodes are busy.

5. *Transient fluctuation and slight delay would not cause false-positive.* As this experiment is conducted in a distributed environment, delays exist and vary for different nodes when executing the injections. Despite

this intervention, our framework still does not report transient fluctuations and slight delays as anomalies.

Based on the evaluation results, we find that our solution is able to correctly identify all the anomalies in all these 5 different cases.

5.3 Results Comparison

To demonstrate the superiority of our framework, we also conduct experiments to identify the anomalies with the same injection settings using the alternative methods including *contextual anomaly detection (CAD)* and *rule-based continuous query (Rule-CQ)*. The contextual anomaly detection is equivalent to the snapshot scoring in our framework. For the rule-based continuous query, we define three rules to capture three types of anomalies, including high CPU utilization (rule 1), low CPU utilization (rule 2), and high memory usage anomalies (rule 3), respectively. Different combinations of the three rules are used in the experiments.

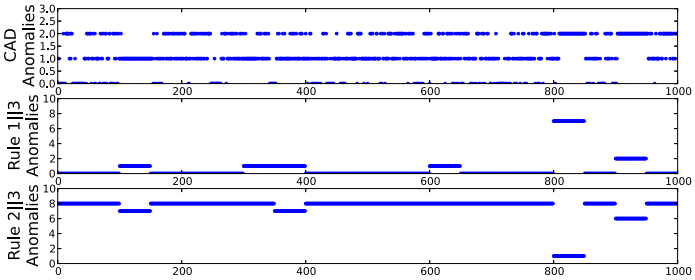


Figure 8: Generated alerts by CAD and Rule-CQ

The generated alerts of these methods are shown in Figure 8, where the x-axis denotes the time and y-axis denotes the number of anomalies. As illustrated, the contextual anomaly detection method generates a lot of false alerts. This is because this method is sensitive to the transient fluctuation. Once an observation deviates from the others at a timestamp, an alert would be triggered. For Rule-CQ method, we experiment all the combinations and report the results of the two best combinations: C1 (rule 1 or rule 2) and C2 (rule 2 or rule 3). Similarly, the Rule-CQ method also generates many false alerts since it is difficult to use rules to cover all the anomaly situations. Table 2 quantitatively shows the precision, recall, and F-measure of the three methods as well as the results of our method. The low-precision and high-recall results of CAD and Rule-CQ indicate that all these method are too sensitive to fluctuations.

Table 2: Measures of different methods performed on the trace data in distributed environment

Measure \ Method	precision	recall	F-measure
CAD	0.4207	1.0000	0.5922
C1: Rule 1||3	0.5381	1.0000	0.6997
C2: Rule 2||3	0.0469	1.0000	0.0897
Our method (worst case)	0.9832	0.8400	0.9060

5.4 A real system problem detected

We have identified a real system problem when deployed our framework on two computing clusters in our department. In one of the clusters, we continuously receive alerts. Logging into the cluster, we find the CPU utilization is high even no tasks are running. We further identify that the high CPU utilization is caused by several processes named *hfsd*. We reported the anomaly to IT support staffs and they confirmed that there exist some problems in this cluster. The high CPU utilization is caused by continuous attempts to connect to a failure node in the network file system. After fixing this problem, these out-of-expectation but real alerts disappear.

6. RELATED WORKS

Although anomaly detection has been studied for years [15, 8]. To the best of our knowledge, our method is the first one that focused on mining contextual collective anomalies among multiple streams in real time. In this section, we briefly review two closely related areas: mining outliers from data streams and mining outliers from trajectories.

With the emerging requirements of mining data streams, several techniques have been proposed to handle the data incrementally [33, 20, 19, 28, 18]. Pokrajac et al. [25] modified the static Local Outlier Factor (LOF) [6] method as an incremental algorithm, and then applied it to find data instance anomalies from the data stream. Takeuchi and Yamanishi [16] trained a probabilistic model with an online discounting learning algorithm, and then use the training model to identify the data instance anomalies. Angiulli and Fassetti [3] proposed a distance-based outlier detection algorithm to find the data instance anomalies over the data stream. However, all the aforementioned works focused on the anomaly detection of a single stream, while our work is designed to discover the contextual collective anomalies over multiple data streams.

A lot of works have been conducted on trajectory outlier detection. One of the representative work on trajectory outlier detection is conducted by Lee et al. [21]. They proposed a partition-and-detection framework to identify the anomaly sub-trajectory via the distance-based measurement. Liang et al. [29] improved the efficiency of Lee's work by only computing the distances among the sub-trajectories in the same grid. As the aforementioned two algorithms require to access the entire dataset, they cannot be adapted to trajectory streams. To address the limitation, Bu et al. [7] proposed a novel framework to detect anomalies over continuous trajectory streams. They built local clusters for trajectories and leveraged efficient pruning strategies as well as indexing to reduce the computational cost. However, their approach identified anomalies based on the local-continuity property of the trajectory, while our method does not make such an assumption. Our approach is close to the work of Ge et al. [13], where they proposed an incremental approach to maintain the top-K evolving trajectories for traffic monitoring. However, their approach mainly focused on the geospatial data instances and ignored the temporal correlations, while our approach explicitly considers the temporal information of the data instances.

7. CONCLUSION

In this paper, we propose a real time anomaly detection framework to identify the contextual collective anomalies from a collection of streams. Our proposed method firstly quantifies the snapshot level anomaly of each stream based on the contextual information. Then the contextual information and the historical information are used in combina-

tion to quantify the anomaly severity of each stream. Based on the distribution of the stream anomaly scores, an implicit threshold is dynamically calculated and the alerts are triggered accordingly. To demonstrate the usefulness of the proposed framework, several sets of experiments are conducted to demonstrate its effectiveness and efficiency.

8. ACKNOWLEDGEMENT

The work was supported in part by the National Science Foundation under grants DBI-0850203, HRD-0833093, CNS-1126619, and IIS-1213026, the U.S. Department of Homeland Security under grant Award Number 2010-ST-06200039, Army Research Office under grant number W911NF-10-1-0366 and W911NF-12-1-0431, National Natural Science Foundation of China under grant number 61300053, and an FIU Dissertation Year Fellowship.

9. REFERENCES

[1] A.Arasu, B.Babcock, S.Babu, M.Datar, K.Ito, I.Nizhizawa, J.Rosenstein, and J.Widom. Stream: The stanford stream data manager. In *SIGMOD*, 2003.

[2] J. Agrawal, Y. Diao, D. Gyllstrom, and N. Immerman. Efficient pattern matching over event streams. In *SIGMOD*, 2008.

[3] F. Anguilli and F. Fassetti. Detecting distance-based outliers in streams of data. In *CIKM*, 2007.

[4] Y. Bai, F. Wang, P. Liu, C. Zaniolo, and S. Liu. Rfid data processing with a data stream query language. In *ICDE*, 2007.

[5] M. Blum, R. Floyd, V.Pratt, R.Rivest, and R. Tarjan. Time bounds for selection. *Journal of Computer System Science*, 1973.

[6] M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. Lof: Identifying density-based local outliers. In *SIGMOD*, 2000.

[7] Y. Bu, L. Chen, A. W.-C. Fu, and D. Liu. Efficient anomaly monitoring over moving object trajectory streams. In *KDD*, 2009.

[8] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 2009.

[9] S. Chandrasekaran, O.Cooper, A.Deshpande, M.J.Franklin, J.M.Hellerstein, W.Hong, S.Krishnamurthy, S.R.Madden, V.Raman, F.Reiss, and M.A.Shah. Telegraphcq: Continous dataflow processing for an uncertain world. In *CIDR*, 2003.

[10] D.Carney, U.Cetintemel, M.Cherniack, C. Convey, S.Lee, G.Seidman, M.Stonebraker, N.Tatbul, and S.Zdonik. Monitoring streams: A new class of data management applications. In *VLDB*, 2002.

[11] J. Dean and S. Ghemawat. Mapreduce: Simplified data processing on large clusters. In *OSDI*, 2004.

[12] R. A. Fisher, F. Yates, et al. Statistical tables for biological, agricultural and medical research. *Statistical tables for biological, agricultural and medical research.*, (Ed. 3.), 1949.

[13] Y. Ge, H. Xiong, Z.-H. Zhou, H. Ozdemir, J. Yu, and K. C. Lee. Top-eye: Top-k evolving trajectory outlier detection. In *CIKM*, 2010.

[14] M. Gupta, A. B. Sharma, H. Chen, and G. Jiang. Context-aware time series anomaly detection for complex systems. In *WORKSHOP NOTES*, page 14, 2013.

[15] V. Hodge and J. Austin. A survey of outlier detection methodologies. *Artificial Intelligence Review*, 2004.

[16] J. ichi Takeuchi and K. Yamanishi. A unifying framework for detecting outliers and change points from time series. *IEEE Transactions on Knowledge and Data Engineering*, 2006.

[17] G. Jiang, H. Chen, and K. Yoshihira. Modeling and tracking of transaction flow dynamics for fault detection in complex systems. *Dependable and Secure Computing, IEEE Transactions on*, 3(4):312–326, 2006.

[18] Y. Jiang, C.-S. Perng, T. Li, and R. Chang. Asap: A self-adaptive prediction system for instant cloud resource demand provisioning. In *ICDM*, 2011.

[19] Y. Jiang, C.-S. Perng, T. Li, and R. Chang. Intelligent cloud capacity management. In *NOMS*, 2012.

[20] Y. Jiang, C.-S. Perng, T. Li, and R. Chang. Self-adaptive cloud capacity planning. In *SCC*, 2012.

[21] J.-G. Lee, J. Han, and X. Li. Trajectory outlier detection: A partition-and-detect framework. In *ICDE*, 2008.

[22] B. Lo, S. Thiemjarus, R. king, and G. Yang. Body sensor network-a wireless sensor platform for pervasive healthcare monitoring. In *PERVASIVE*, 2005.

[23] B. Mozafari, K. Zeng, and C. Zaniolo. High-performance complex event processing over xml streams. In *SIGMOD*, 2012.

[24] A. Oliner, A. Ganapathi, and W. Xu. Advances and challenges in log analysis. *Comm. of ACM*, 2012.

[25] D. Pokrajac, A. Lazarevic, and L. J. Latecki. Incremental local outlier detection for data streams. In *CIDM*, 2007.

[26] Y. Song and L. Cao. Graph-based coupled behavior analysis: a case study on detecing collaborative manipulations in stock markets. In *IEEE world congress on computational intelligence*, 2012.

[27] storm. http://storm-project.net.

[28] L. Tang, C. Tang, L. Duan, Y. Jiang, C. Zeng, and J. Zhu. Movstream: An efficient algorithm for monitoring clusters evolving in data streams. In *Grc*, 2008.

[29] L. Tang, C. Tang, Y. Jiang, C. Li, L. Duan, C. Zeng, and K. Xu. Troadgrid: An efficient trajectory outlier detection algorithm with grid-based space division. In *NDBC*, 2008.

[30] L. G. Valiant. A bridging model for parallel computation. *Communications of the ACM*, 33(8):103–111, 1990.

[31] C. Zeng, Y. Jiang, L. Zheng, J. Li, L. Li, H. Li, C. Shen, W. Zhou, T. Li, B. Duan, et al. Fiu-miner: a fast, integrated, and user-friendly system for data mining in distributed environment. In *KDD*, pages 1506–1509. ACM, 2013.

[32] L. Zheng, C. Shen, L. Tang, T. Li, S. Luis, S.-C. Chen, and V. Hristidis. Using data mining techniques to address critical information exchange needs in disaster affected public-private networks. In *KDD*, 2010.

[33] Y. Zhu and D. Shasha. Statstream: Statistical monitoring of thousands of data streams in real time. In *VLDB*, 2002.

TerraFly GeoCloud: An Online Spatial Data Analysis and Visualization System

Mingjin Zhang, Florida International University
Huibo Wang, Florida International University
Yun Lu, Florida International University
Tao Li, Florida International University
Yudong Guang, Florida International University
Chang Liu, Florida International University
Erik Edrosa, Florida International University
Hongtai Li, Florida International University
Naphtali Rishe, Florida International University

With the exponential growth of the usage of web map services, the geo data analysis has become more and more popular. This paper develops an online spatial data analysis and visualization system, TerraFly GeoCloud, which facilitates end users to visualize and analyze spatial data, and to share the analysis results. Built on the TerraFly Geo spatial database, TerraFly GeoCloud is an extra layer running upon the TerraFly map and can efficiently support many different visualization functions and spatial data analysis models. Furthermore, users can create unique URLs to visualize and share the analysis results. TerraFly GeoCloud also enables the MapQL technology to customize map visualization using SQL-like statements. The system is available at http://terrafly.fiu.edu/GeoCloud/.

Categories and Subject Descriptors: H.2.8 [**Database Applications**]: Data mining, Spatial databases and GIS

General Terms: Design, Algorithms, Performance

Additional Key Words and Phrases: Geospatial analysis, GIS, Visualization, Big Data

1. INTRODUCTION

With the exponential growth of the World Wide Web, there are many domains, such as water management, crime mapping, disease analysis, and real estate, open to Geographic Information System (GIS) applications. The Web can provide a giant amount of information to a multitude of users, making GIS available to a wider range of public users than ever before. Web-based map services are the most important application of modern GIS systems. For example, Google Maps currently has more than 350 million users. There are also a rapidly growing number of geo-enabled applications which utilize web map services on traditional computing platforms as well as the emerging mobile devices.

However, due to the highly complex and dynamic nature of GIS systems, it is quite challenging for end users to quickly understand and analyze the spatial data, and to efficiently share their own data and analysis results to others. First, typical geographic visualization tools are complicated and

This material is based in part upon work supported by the National Science Foundation under Grant Nos. I/UCRC IIP-1338922, AIR IIP-1237818, SBIR IIP-1330943, III-Large IIS-1213026, MRI CNS-0821345, MRI CNS-1126619, CREST HRD-0833093, I/UCRC IIP-0829576, MRI CNS-0959985, FRP IIP-1230661, SBIR IIP-1058428, SBIR IIP-1026265, SBIR IIP-1058606, SBIR IIP-1127251, SBIR IIP-1127412, SBIR IIP-1118610, SBIR IIP-1230265, SBIR IIP-1256641. Includes material licensed by TerraFly (http://terrafly.com) and the NSF CAKE Center (http://cake.fiu.edu).
Author's addresses: M. Zhang, H. Wang, Y. Lu, T. Li, Y. Guang, E. Edrosa, H. Li, N. Rishe, School of Computing and Information Sciences, Florida International University; 11200 SW 8th Street, Miami, FL, 33199, USA.

DOI : http://dx.doi.org/10.1145/0000000.0000000

fussy with a lot of low-level details, thus they are difficult to use for spatial data analysis. Second, the analysis of large amount spatial data is very resource-consuming. Third, current spatial data visualization tools are not well integrated for map developers and it is difficult for end users to create the map applications on their own spatial datasets.

To address the above challenges, this paper presents TerraFly GeoCloud, an online spatial data analysis and visualization system, which allows end users to easily visualize and analyze various types of spatial data. TerraFly GeoCloud offers the following important features to facilitate the spatial data analysis.

— First, TerraFly GeoCloud can accurately visualize and manipulate point and polygon spatial data with just a few clicks.
— Second, TerraFly GeoCloud employs an analysis engine to support the online analysis of spatial data, and the visualization of the analysis results. Many different spatial analysis functionalities are provided by the analysis engine.
— Third, based on the TerraFly map API, TerraFly GeoCloud offers a MapQL language with SQL-like statements to execute spatial queries, and render maps to visualize the customized query results.

Our TerraFly GeoCloud online spatial data analysis and visualization system is built upon the TerraFly system using TerraFly Maps API and JavaScript TerraFly API add-ons in a high performance cloud Environment. The function modules in the analysis engine are implemented using C and R language and python scripts. Comparing with current GIS applications, our system is more user-friendly and offers better usability in the analysis and visualization of spatial data. The system is available at http://terrafly.fiu.edu/GeoCloud/.

A preliminary version of the work focusing on visualization solutions (e.g., map rendering and spatial data visualization) is published in [Lu et al. 2013a]. In this journal submission, we added many spatial analysis functions and also made the result visualization more interactive. With these changes TerraFly Geocloud became more intelligent and can be applied in many application domains, such as disease analysis, crime analysis, and real estate analysis. We present several application case studies including Florida property analysis and Lung cancer analysis to demonstrate the usefulness of the system.

In summary, the TerraFly GeoCloud system is a type of intelligent decision support system. By leveraging distributed computing, map rendering, visualization technologies, and spatial data mining techniques, TerraFly GeoCloud enables users to perform different types of spatial data analysis tasks for decision support (e.g., gathering and analyzing data, identifying/diagnosing problems, proposing possible actions and strategies, and evaluating the proposed actions and strategies) [Matsatsinis and Siskos 2003]. Analysis functions supported in TerraFly GeoCloud include spatial data visualization, spatial dependency and auto-correlation, spatial data clustering, spatial regression, measuring geographic distribution, spatial interpolation, and customize map visualization. It also leverages rich user interactions to perform data analysis and support human decision intelligently. Two real case studies including Florida property analysis and Lung Cancer analysis using GeoCloud shows how TerraFly GeoCloud helps user perform data analysis and visualization to make decisions. The rest of this paper is organized as follows: Section 2 describes the architecture and the system overview of TerraFly GeoCloud; Section 3 describes the visualization and analysis methods in TerraFly GeoCloud; Section 4 describes the MapQL spatial query language and customized map visualization with MapQL; Section 5 studies the system performance for both on-line and off-line analysis; Section 6 presents the case studies on the online spatial analysis; Section 7 discusses the related work; and finally Section 8 concludes the paper.

2. SYSTEM OVERVIEW

TerraFly GeoCloud is built upon the TerraFly system to support various kinds of online spatial data analysis using TerraFly Maps API and TerraFly API add-ons in a high performance cloud Environ-

ment. We first introduce the TerraFly system and then describe the overall system demonstration of GeoCloud.

2.1. TerraFly

TerraFly is a system for querying and visualizing of geospatial data developed by High Performance Database Research Center (HPDRC) lab in Florida International University (FIU). This TerraFly system serves worldwide web map requests over 125 countries and regions, providing users with customized aerial photography, satellite imagery and various overlays, such as street names, roads, restaurants, services and demographic data [Rishe et al. 2001; Rishe et al. 2005].

TerraFly allows users to virtually fly over enormous geographic information simply via a web browser with a bunch of advanced functionalities and features such as user-friendly geospatial querying interface, map display with user-specific granularity, real-time data suppliers, demographic analysis, annotation, route dissemination via autopilots and API for web sites, etc. TerraFly's server farm ingests geolocates, mosaics, and cross-references 40TB of base map data and user-specific data streams.

2.2. TerraFly GeoCloud

Figure 1 shows the system architecture of TerraFly GeoCloud. Based on the current TerraFly system including the Map API and all sorts of TerraFly data, we developed the TerraFly GeoCloud system to perform online spatial data analysis and visualization. In TerraFly GeoCloud, users can import and visualize various types of spatial data (data with geo-location information) on the TerraFly map, edit the data, perform spatial data analysis, and visualize and share the analysis results to others. Available spatial data sources in TerraFly GeoCloud include but not limited to demographic census, real estate, disaster, hydrology, retail, crime, and disease. In addition, the system supports MapQL, which is a technology to customize map visualization using SQL-like statements.

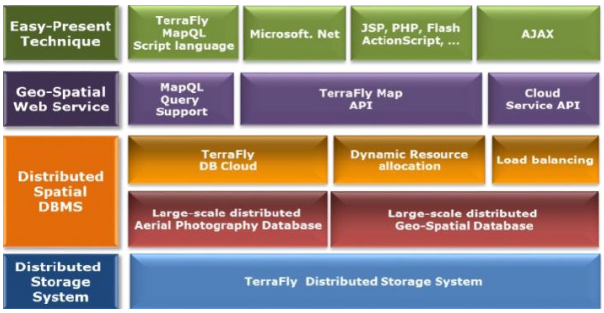


Fig. 1: The Architecture of TerraFly GeoCloud

The spatial data analysis functions provided by TerraFly GeoCloud include spatial data visualization (visualizing the spatial data), spatial dependency and autocorrelation (checking for spatial dependencies), spatial clustering (grouping similar spatial objects),spatial regression, measuring Geographic Distribution and Kriging (geo-statistical estimator for unobserved locations).

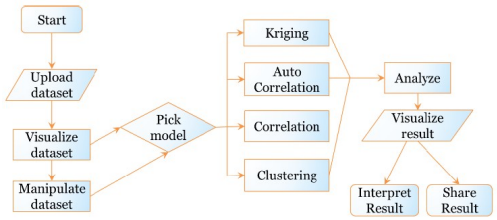
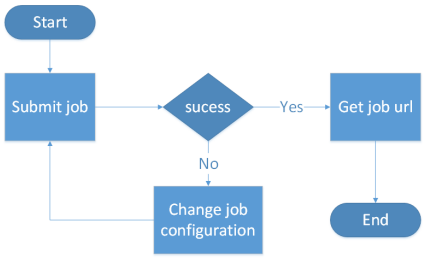
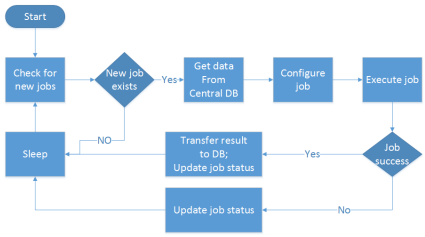


Fig. 2: The Workflow of TerraFly Geocloud

Figure 2 shows the data analysis workflow of the TerraFly GeoCloud system. Users first upload datasets to the system, or view the available datasets in the system. User can upload GeoJson, Shapefile and .asc file. They can then visualize the data sets with customized appearances. By Manipulating the dataset, users can edit the dataset and perform pre-processing (e.g., adding more columns). Followed by pre-processing, users can choose proper spatial analysis functions and perform the analysis. After the analysis, they can visualize the results and also share them with others.



(a) The front-end workflow of offline analysis



(b) The back-end workflow of offline analysis

Fig. 3: The workflow of offline analysis

Geocloud also supports offline analysis, if users want to perform analysis on large data sets. Figure 3 shows the workflow of the offline analysis in TerreFly GeoCloud. The workflow in the front-end is shown in Figure 3a. Users can submit jobs through the GeoCloud website. If the job submission failed, users should change the job configurations. If a job is accepted successfully, the user will receive a URL from which the analysis results can be downloaded. The offline job status can be shown through the URL. Figure 3b shows the back-end workflow of the offline analysis. The system polls the database for new jobs. If a new job exists, first, the system will retrieve data from the central DB. Second, the system will configure the job using the submitted configuration. Third, the system will copy the data to HDFS, send the job to the GeoCloud hadoop platform, and run the hadoop job. If the job is successfully completed, the results will be transferred to the database. After the jobs status being updated, users can download the analysis results through the URL.

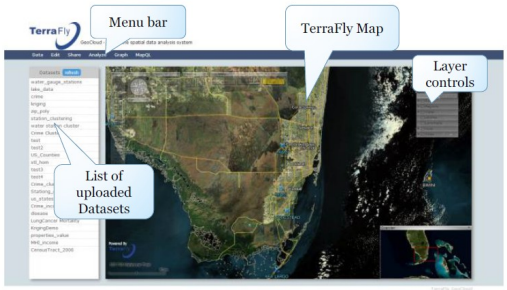


Fig. 4: Interface of TerraFly Geocloud

Figure 4 shows the interface of the TerraFly GeoCloud system. The top bar is the menu of all functions, including Data, analysis, Graph, Share, and MapQL. The left side shows the available datasets, including both the uploaded datasets from the user and the existing datasets in the system. The right map is the main map from TerraFly. This map is composed by TerraFly API, and it includes a detailed base map and diverse overlays which can present different kinds of geographical data.

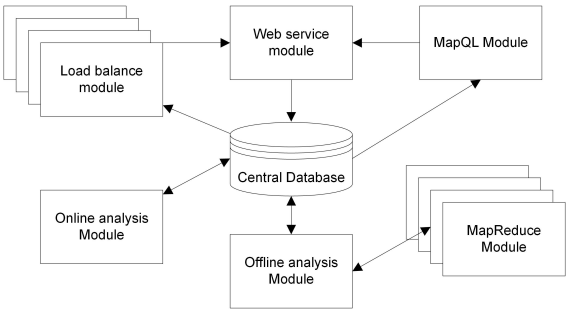


Fig. 5: Modules of the GeoCloud system

Figure 5 shows the main function modules of the GeoCloud system. The center of the system is a central database which holds all the system related data. The central database composed by the sksOpen database, the map file database, and the relational databases such as SQL Server and PostGreSQL. The sksOpen database is a spatial object hybrid index and storage system that includes both an R-Tree spatial index and an inverted text file index, which attained fast retrieval of spatial data even when the matching objects were located far away from one another [Lu et al. 2013b]. The map file database provides the base map for users, and the relational databases are used for storing the uploaded data and the analysis results. The online and offline analysis modules process the analysis tasks and push back the results to the Central Database. The online analysis module processes analysis tasks which can be done at runtime while the offline analysis module employs the MapReduce module to process heavy duty tasks. The load balance module and web service module leverage distributed spatial data visualization with autonomic resource management techniques to provide the on-demand and balanced resource allocation to achieve the QoS (Quality of service).

TerraFly GeoCloud also provides MapQL spatial query and render tools. MapQL supports SQL-like statements to realize the spatial query, and render the map according to users inputs. MapQL tools can help users visualize their own data using a simple statement. This provides users with a

better mechanism to easily visualize geographical data and analysis results. Shown in Figure 5, the MapQL module creates map visualization at runtime based on the MapQL statements.

3. VISUALIZATION AND ANALYSIS METHODS

Many different visualization functions and spatial data analysis models are provided in TerraFly GeoCloud. TerraFly GeoCloud also integrates spatial data mining and data visualization. The spatial data mining results can be easily visualized. In addition, visualization can often be incorporated into the spatial mining process.

3.1. Spatial Data Visualization

Fig. 6: Spatial Data Visualization: Point data and Polygon Data

For spatial data visualization, the system supports both point data and polygon data and users can choose color or color range of data for displaying. As shown in Figure 6, the point data is displayed on left, and the polygen data is displayed on the right. The data labels are shown on the base map as extra layers for point data, and the data polygons are shown on the base map for polygon data. Many different visualization choices are supported for both point data and polygon data. For point data, users can customize different parameters such as the icon style, icon color or color range, and label value. For polygon data, users can customize different parameters including the fill color or color range, fill alpha, line color, line width, line alpha, and label value.

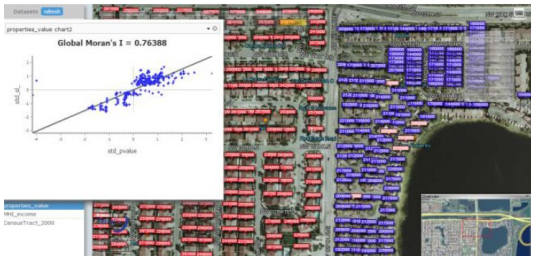
3.2. Spatial Dependency and Auto-Correlation

Spatial dependency is the co-variation of properties within the geographic space: characteristics at proximal locations that appear to be correlated, either positively or negatively. Spatial dependency leads to the spatial autocorrelation problem in statistics [De Knegt et al. 2010]. Spatial autocorrelation is more complex than one-dimensional autocorrelation because spatial correlation is multi-dimensional and multi-directional. The TerraFly GeoCloud system provides auto-correlation analysis tools to discover spatial dependencies in a geographic space, including global and local clusters analysis where Moran's I measure is used [Li et al. 2007]. Formally, Morans I, the slope of the line, estimates the overall global degree of spatial autocorrelation as follows:

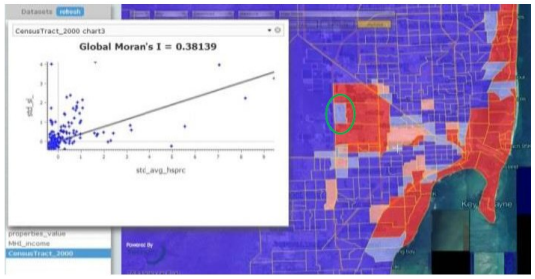
$$I = \frac{n}{\sum_i^n \sum_j^n w_{ij}} * \frac{\sum_i^n \sum_j^n w_{ij}(y_i - \hat{y})(y_i - \hat{y})}{\sum_i^n (y_i - \hat{y})^2}, \tag{1}$$

where w_{ij} is the weight, $w_{ij} = 1$ if locations i and j are adjacent and zero otherwise $w_{ii} = 0$ (a region is not adjacent to itself).y_i and $\hat{y}$ are the variable in the i-th location and the mean of the variable, respectively. n is the total number of observations. Morans I is used to test hypotheses concerning the correlation, ranging between 1.0 and $+1.0$. Morans I measures can be displayed as a checkerboard where a positive Morans I measure indicates the clustering of similar values and a negative Morans I measure indicate dissimilar values. TerraFly GeoCloud provides auto-correlation analysis tools to check for spatial dependencies in a geographic space, including global and local clusters analysis.

Figure 7b shows an example of spatial auto-correlation analysis on the average properties price by zip code data in Miami (polygondata). Each dot here in the scatterplot corresponds to one zip code. The first and third quadrants of the plot represent positive associations (high-high and low-low), while the second and fourth quadrants represent associations (low-high, high-low). For example, the green circle area is in the low-high quadrants. The density of the quadrants represents the dominating local spatial process. The properties in Miami Beach are more expensive, and are in the high-high area. Figure 7a presents the auto-correlation analysis results on the individual properties price in



(a) Properties value in Miami



(b) Average properties price by zip code in Miami

Fig. 7: Spatial Dependency and Auto-Correlation

Miami (point data). Each dot here in the scatterplot corresponds to one property. As the figure shows, the properties near the big lake are cheaper, while the properties along the west are more expensive.

3.3. Spatial Data Clustering

Spatial data clustering algorithms identify clusters, or densely populated regions, according to some distance measures in a large, multidimensional dataset. Several spatial clustering techniques are provided in TerraFly GeoCloud.

K-Means. K-means is an efficient clustering algorithm. K-means partition all the data set in to k cluster. Firstly, the algorithm will randomly find k initial center points. Secondly, finding the nearest center point for each record as its cluster and getting mean value for each cluster as new cluster center. Repeating first and second step until the cluster center doesn't change. In TerraFly GeoCloud system, user can apply k-means algorithm by inputing cluster number.

DBSCAN. The TerraFly GeoCloud system supports the DBSCAN (for density-based spatial clustering of applications with noise) data clustering algorithm [Ester et al. 1996]. DBSCAN is a density-based clustering algorithm and it finds a number of clusters starting from the estimated density distribution of corresponding nodes. DBSCAN requires two parameters as the input: eps (the neighbor size) and minPts (the minimum number of points required to form a cluster). It starts with

an arbitrary starting point that has not been visited so far. This point's neighborhood is retrieved, and if it contains sufficiently many points, a cluster is started. Otherwise, the point is labeled as a noise point [Ester et al. 1996]. If a point is found to be a dense part of a cluster, its neighborhood is also part of that cluster. Hence, all points that are found within the neighborhood are added. This process continues until the density-connected cluster is completely identified. Then, a new unvisited point is retrieved and processed, leading to the discovery of new cluster or noise points [Bilodeau et al. 2005]. Figure 8a shows an example of DBSCAN clustering on the crime data in Miami. As shown in Figure 8a, each point is an individual crime record marked on the place where the crime happened, and the number displayed in the label is the crime ID. By using the clustering algorithm, the crime records are grouped, and different clusters are represented by different colors on the map.

Cluster Detection. Kulldorff & Nagarwalla(KN)[Kulldorff 1997] provides a method to perform cluster detection. KN method is implemented by scanning all the area using circular zones of variable size. KN method is widely used in spatial epidemiology. The steps of KN method include: (1). Move a circle in space to obtain an infinite number of overlapping circles; (2). Compute LLR (Log Likelihood Ratio) of each circle and sort the LLR; and (3). Get some large LLR then use Monte Carlo method to calculate P-value of them. The Log Likelihood Ratio can be calculated as follow:

$$LLR = max_j(\frac{Y_j}{E_j})^{Y_j}(\frac{Y_+ + Y_j}{Y_+ - E_j})^{Y_+ - Y_J} I(Yj > Ej), \tag{2}$$

where Y_j denotes the observed number of instance in circle area, $Y+$ denotes the number of instance in all the area, E_j denotes the expected number of instance in circle area. Figure 8b shows the result of lung cancer cluster map in Florida. The red points indicate the disease cluster where the unusual disease case happened. The number in the red point is the p-value of each area.[Elliott and Wartenberg 2004]

HotSpot. HotSpot analysis function using Gi* statistic method aims to detect the hot (or cold) cluster which has a high (or a low) Gi* value. Figure 8c shows the result of the hotspot cluster map of lung cancer mortality in Florida. From this map, we can observe that the central part which is covered by red color is a hot cluster and four counties in the south region forms a cold cluster.

Outlier Analysis. Outlier analysis recognizes the outliers whose attributes values are different from their neighbors. In TerraFly GeoCloud, local moran's I map, z-value map, and p-value map are provided.

3.4. Spatial Regression

Regression tools can be used to estimate relationships between attributes.

Linear Regression. TerraFly GeoCloud provides linear regression tools with multiple tests, such as global morans I test. Figure 9a shows the linear regression results between mortality and median house price and median income. It should be noted that global Morans I test indicates that the residual is geo-correlated, and thus linear regression model is not a good fit for this problem.

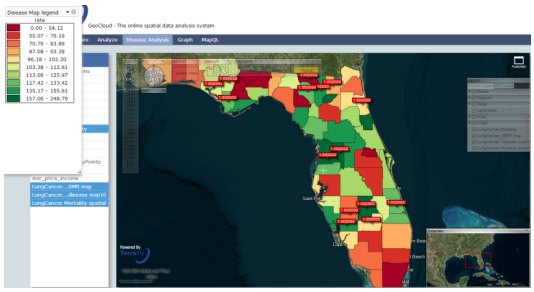
Spatial auto-regression. In spatial auto-regression, a lag model and an error model are provided. The spatial auto-regression lag model can be calculated as follows:

$$Y = \rho Wy + x\beta + \epsilon, \tag{3}$$

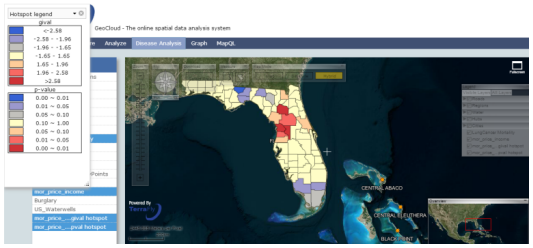
where Y is a dependent variable, W is a matrix of spatial weights, x is an independent variable, β denotes the unknown parameters, and ϵ is an error term.

Figure 9b shows the result of a spatial auto-regression lag model. In this model, multiple test methods are provided for verifiability: Wald test is used to determine whether various parameters can be zero or not; AIC for linear regression and lag model is applied to indicate which model is better; LR test, the Likelihood Ratio diagnostics, is used for testing spatial dependence; and LM test

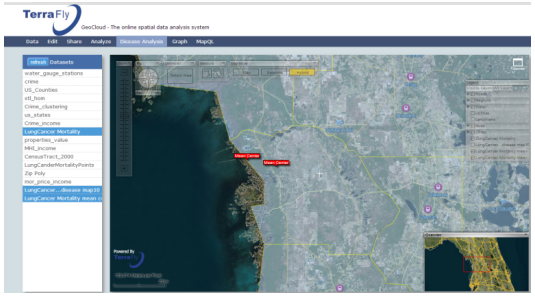
(a) DBSCAN clustering on the crime data in Miami



(b) KN cluster detection on lung cancer in Florida



(c) Hotspot clustering on lung cancer in Florida



(d) Center point and weighted center point

Fig. 8: Spatial Clustering in Geocloud

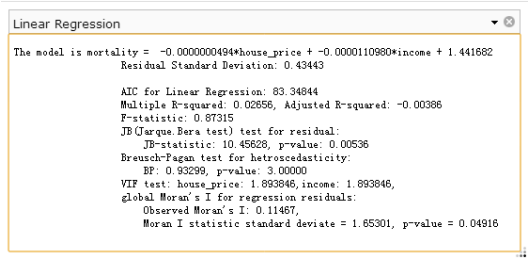
is utilized for evaluating the absence of spatial autocorrelation in lag model residuals [Dubin et al. 1999][Kelejian and Prucha 1998].

3.5. Measuring Geographic Distribution

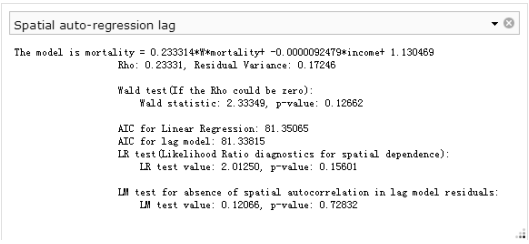
Geographic distribution measurements include mean/median central, standard distance, and distributional trends functions. In our system, a weighted mean central is provided as follow:

$$X = \frac{\Sigma_i w_i x_i}{\Sigma_i w_i}, Y = \frac{\Sigma_i w_i y_i}{\Sigma_i w_i}, \tag{4}$$

where x_i and y_i denote the coordinate of each point (but when the data set is polygonal, x_i and y_i indicate the center of each polygon) and w_i is the weight which corresponds in our system to mortality or incidence. Figure 8d shows these two type of points: one type is the non-weighted center point, and the other type is the lung cancer mortality weighed center point. Besides the center/median point function, TerraFly GeoCloud also includes distributional trends and standard distance.



(a) Linear regression tool on lung cancer in Florida



(b) Spatial auto-regression lag model on lung cancer in Florida

Fig. 9: Spatial Regression in Geocloud

3.6. Spatial Interpolation Method

Kriging is a geo-statistical estimator that infers the value of a random field at an unobserved location (e.g. elevation as a function of geographic coordinates) from samples (see spatial analysis) [Stein 1999] Figure 10 shows an example of Kriging. The data set is the water level from water stations

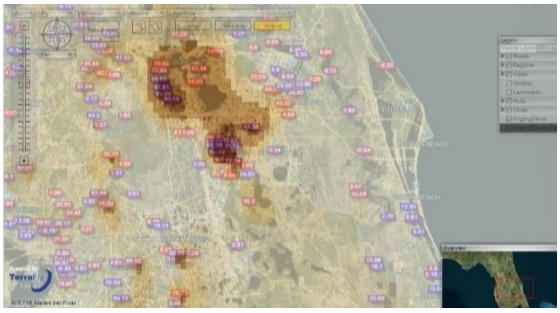


Fig. 10: Kriging data of the water level in Florida

in central Florida. Note that not all the water surfaces are measured by water stations. The Kriging results are estimates of the water levels and are shown by the yellow layer.

4. CUSTOMIZED MAP VISUALIZATION

TerraFly GeoCloud also provides MapQL spatial query and render tools, which supports SQL-like statements to facilitate the spatial query and more importantly, render the map according users requests. This is a better interface than API to facilitate developer and end user to use the TerraFly map as their wish. By using MapQL tools, users can easily create their own maps.

4.1. Introduction and Implementation

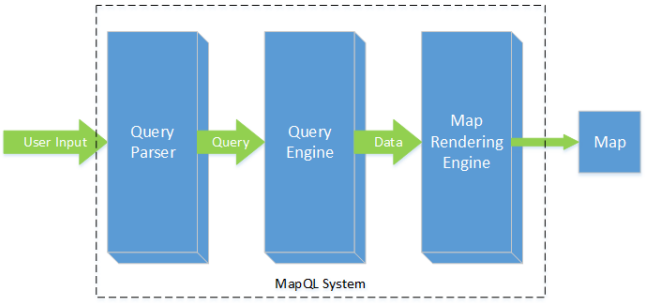


Fig. 11: MapQL System Architecture

MapQL is an extension of GeoSPARQL, which is a standard for representation and querying of geospatial linked data. MapQL defined some new key words that include T_ICON_PATH, T_LABEL, T_LABEL_SIZE, T_FILED_COLOR, T_THICKNESS, T_OPACITY and T_BORDER_COLOR to facilitate customized map visualization. The architecture of MapQL is shown in Figure 11. MapQL contains three modules: Query parser, Query Engine, and Map Rendering Engine. Query Parser checks syntax and semantic correctness of the input query. After passing Query Parser, the query goes to Query Engine where it is committed to the database. The PostGreSQL database, which has a very good support for spatial data indexing and query, is used in the Query Engine module. The returned results from Query Engine will be processed at Map Rendering Engine. Mapnik, a toolkit for making customized map, is used in Map Rendering Engine to create customized maps and put them as a layer on TerraFly map through TerraFly map API. The workflow of MapQL is shown in Figure 12. The input of the whole procedure is MapQL statements, and the output is map visualization rendered by the MapQL engine.

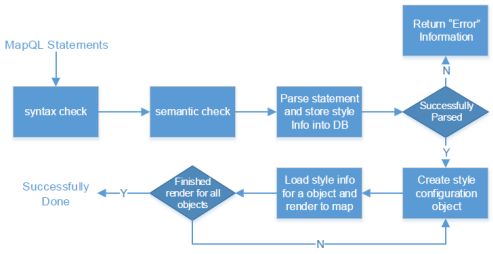


Fig. 12: The workflow of MapQL

Shown in Figure 12, the first step is the syntax check of the statements. The syntax check guarantees that the syntax of an input query conforms to the standard (e.g., the spelling-check of the reserved words). The semantic check ensures that the data source name and metadata which MapQL statements want to visit are correct. After the above two checks, the system will parse the statements and store the parse results including the style information into a spatial database. The style information includes where to render and what to render. After all the style information is stored, the system will create style configuration objects for rendering. The last step is for each object, load the style information from the spatial database and render to the map according to the style information.

We implemented the MapQL tools using C++. For the last step of rendering the objects to the map visualization, we employed the TerraFly map render engine.

4.2. Query Examples

For example, if we want to query the house prices near Florida International University, we use MapQL statements in Figure 13. There are four reserved words in the statements, T_ICON_PATH ,

```
SELECT
    '/var/www/cgi-bin/house.png' AS T_ICON_PATH,
r.price AS T LABEL,
    '15' AS T LABEL SIZE,
r.geo AS GEO
FROM
realtor 20121116 r
WHERE
ST Distance(r.geo,  GeomFromText('POINT(-80.376283 25.757228)')) <
0.03;
```

Fig. 13: Query house prices using MapQL

T_LABEL, T_LABEL_SIZE , and GEO. We use T_ICON_PATH to store the customized icon. Here we choose a local png file as icon. T_LABEL denotes that icon label that will be shown on the map, T_LABEL_SIZE is the pixel size of the label, and GEO is the spatial search geometry. The statements go through the syntax check first. If there is incorrect usage of reserved words or wrong spelling of the syntax, the statements will be corrected or Error information will be sent to the user. For example, if the spelling of select is not correct, Error information will be sent to the user. The semantic check makes sure that the data source name realtor_20121116 and metadata r. price and r.geo are exist and available. After the checks, the system parsed the statements. The SQL part will return corresponding results including the locations and names of nearby objects, the MapQL part will collect the style information including icon path and icon label style. Both of them are stored into a spatial database. The system then created style configuration objects for query results. The last step is rendering all the objects on the map visualizations. The needed style information includes icon picture and label size, and the data information includes label value and location (Lat, Long). Figure 14 shows the result of this query.

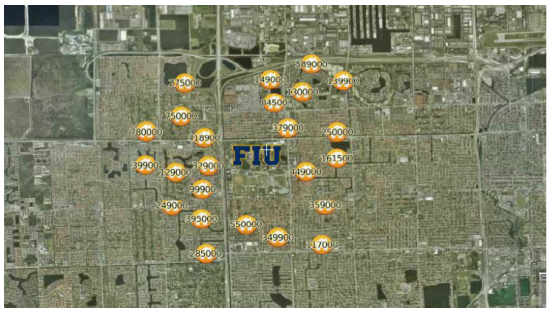


Fig. 14: Result of query house prices using MapQL

In the following, we present several query examples using MapQL statements. Figure 15 shows all the hotels along a certain street within a certain distance and also displays the different stars of the hotels. The MapQL statement for this query is listed below:

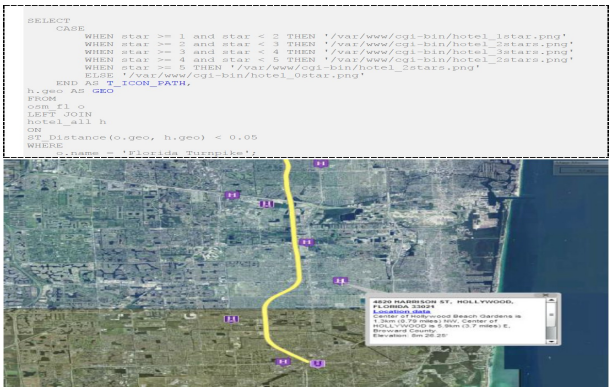


Fig. 15: Query hotel data along the line

Figure 16 shows the traffic of Santiago where the colder the color is, the faster the traffic is; the warmer the color is, the worse the traffic is. The MapQL statement is listed below:

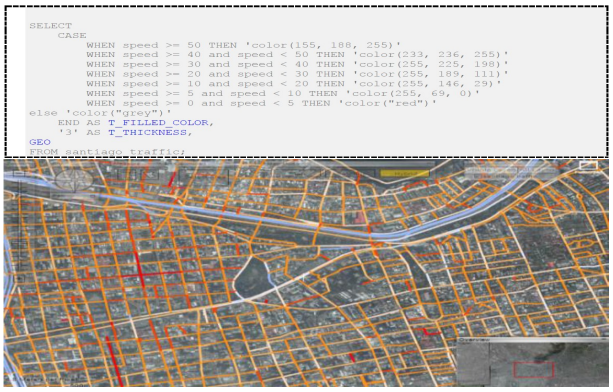


Fig. 16: Query traffic data of Santiago

Figure 17 shows the different average incomes with in different zip codes. In this demo, users can customize the color and style of the map layers, different colors stand for different average incomes. The corresponding MapQL statement is listed below:

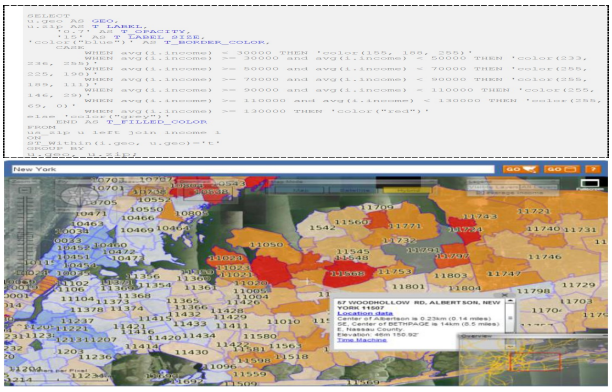


Fig. 17: Query average incomes

All these examples demonstrate that in TerraFly GeoCloud, users can easily create different map applications using simple SQL-like statements.

5. SYSTEM PERFORMANCE

In this section, we evaluate the performance of Terrafly GeoCloud using some example datasets and analysis. Terrafly GeoCloud supports both online analysis and offline analysis. For online analysis, we discuss the performance of correlation analysis; and for offline analysis, we use K-means analysis as an example.

We did not perform performance comparisons with similar products as they typically do not share much about their system design and implementation. Based on the fact that all GeoCloud functions have reasonable running time which facilitated users data analysis, we performed functionality comparisons with products whose functions are available (such as GeoDa and ArcGis [Anselin et al. 2006; Johnston et al. 2001]) in Related Products.

5.1. Online Analysis Performance

The data set used for performance evaluation is Florida_property_value which contains 1,042,281 records, and each record includes longitude, latitude and property value. Figure 18 shows part of the data set on the second zoom level. A yellow point, showing the property value, is used to denote each property.

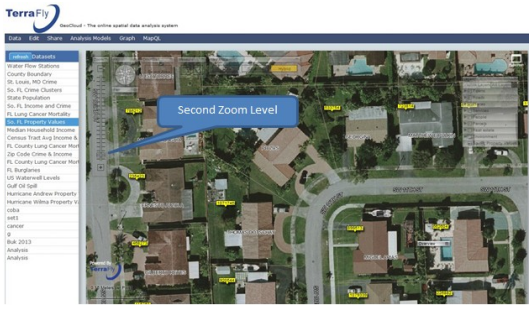


Fig. 18: South Florida Property

In order to provide good user experience, we only show part of the data that can be displayed on user's current screen. When a user zooms out the screen, GeoCloud will load the new data into

the screen and the analysis is then performed on the current displayed data. This guarantees that a user can view the data and obtain the analysis results very quickly. When users want to preform data analysis, most of the time they are more concerned with some local data. For example, if a user wants to buy a property in a certain zip code, and he/she will only care about the property values of his/her interested location. At this time, doing a global analysis is time consuming and unnecessary.

The online analysis performance is related to the zoom level. Here we use the auto-correlation analysis as an example to evaluate the online analysis performance. Figure 19 shows the performance of autocorrelation. The horizontal axis indicates the number of records on each zoom level. The vertical axis denotes the running time. For example, when the user zooms to the third level, there are 52 records showing on the screen, and the autocorrelation analysis needs 0.956s (which includes network communication time, time for analysis, and time for rendering the results on the map) to complete. The time needed for the sixth zoom level is 4 seconds. The sixth zoom level, which contains 1535 data records, is the highest level that all the data can be shown without overlapping. When we zoom to a higher level, too many records are overlapping with each other that makes the results hard to view.

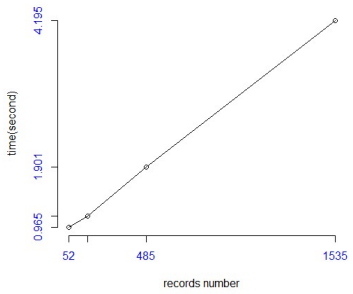


Fig. 19: Auto-Correlation performance

5.2. Offline Analysis Performance

Here we use the K-means clustering method to evaluate offline analysis performance in TerraFly GeoCloud. We apply K-means clustering analysis on Florida_property_value data set. In order to compare the performance of signal machine and hadoop cluster, we duplicate 10 times of the data set, the total number of the records is $19,616,320$.

For the experiment, we set the number of clusters to be 100 and iteration time is 4. The running time for signal machine is 34.83 minutes. Figure 20 shows the running time of hadoop. The vertical axis denotes the running time. The horizontal axis denotes the total task capacity that is the number of cores running parallel, which refer to the total computation power we assigned to the task. When we set total task capacity to 16, the running time of K-means is 7 minutes, so when user wants to perform big data analysis, using Hadoop is more efficient than single machine: when we adding the total task capacity, the performance is increasing, so the running time is decreasing dramatically. Leveraged by the Hadoop platform, we can guarantee the analysis performance by simply adjust total task capacity (computing power).

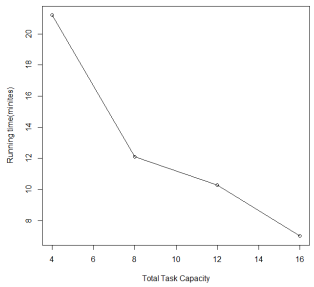


Fig. 20: Parallel K-means performance

6. CASE STUDIES

In this section, we present some case studies on using TerraFly GeoCloud for spatial data analysis and visualization. We use two types of data set, one is Florida property data, the other is Florida Lung cancer mortality to show how to apply Geocloud analysis and visualization function on application domains.

6.1. Florida Property Analysis

As discussed in Section 3.2, we know the results of auto correlation can be shown in a scatter diagram, where the first and third quadrants of the plot represent positive associations, while the second and fourth quadrants represent negative associations. The second quadrant stands for low-high which means the value of the object is low and the values of surrounding objects are high.

A lay user Erik, who has some knowledge about the database and data analysis, wanted to invest a house property in Miami with a good appreciation potential. By using TerraFly GeoCloud, he may obtain some ideas about where to buy. He believes that if a property itself has low price and the surrounding properties have higher values, then the property may have good appreciation potential, and is a good choice for investment. He wants to first identify such properties and then do a field trip with his friends and the realtor agent.

To perform the task, first, Erik checked the average property prices by zip code in Miami which is shown in Figure 7b. He found the green circled area in the low-high quadrants, which means that the average price of properties of this area is lower than the surrounding areas.

So. FL Property Values

id	min_show_level	longitude	latitude	pvalue
9766	1	-80.275203	26.273378	298000
12717	8	-80.142601	26.165103	403000
38997	8	-80.15843	26.340396	381000
44849	8	-80.416315	26.117033	486000
57613	8	-80.42536	26.009357	218000
62752	1	-80.208778	26.184111	149000
73930	8	-80.247377	25.785131	203000
84664	8	-80.11329	26.133841	66000
103612	8	-80.185366	26.101447	189000
106825	8	-80.234288	26.141735	172000
111149	8	-80.31925	26.106385	268000
113091	8	-80.129359	25.835633	1490000

Fig. 21: Sample Data of south_florida_house_price data set

Erik wanted to obtain more insights on the property price in this area. He uploaded a detailed spatial data set named as south_florida_house_price into the TerraFly GeoCloud system.

south_florida_house_price data set contains more than 1 million records and it shows the Geo-location information(coordinates) and price of the property in south Florida. The sample of the data set is shown in Figure 21. He customized the label color range as the properties price changes. And then, he chose different areas in the green circled area in Figure 7b to perform the auto-correlation analysis.

Fig. 22: Properties in Miami

Finally, he found an area shown in Figure 22, where there are some good properties in the low-high quadrants (in yellow circles) with good locations. And one interesting observation is, lots of properties along the road Gratigny Pkwy has lower prices. He was then very excited and wanted to do a query to find all the cheap properties with good appreciation potential along the Gratigny Pkwy. Erik composed the MapQL statements to find out the properties whose distance from the Gratigny Pkwy is less than a threshold and price is lower than the surrounding area, and if the value of the property is between 100,000 to 200,000, using green to denote the property, and if the value between 200,000 and 400,000, using blue to denote the property, and if the value is more than 400,000, using red color to indicate the house.

Fig. 23: MapQL results

The Figure 23 presents the final results of the MapQL statements. Finally, Erik sent the URL of the map visualization out by email, waiting for the response of his friends and the realtor agent.

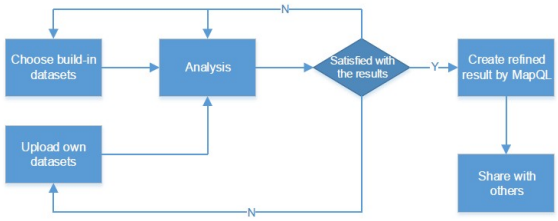


Fig. 24: The flow path of Erik case

Figure 24 illustrates the whole workflow of the case study. In summary, Erik first viewed the system build-in datasets, conducted the data analysis, and then he identified properties of interest. He then composed MapQL statements to create his own map visualization to share with his friends. The case study demonstrates that TerraFly GeoCloud supports the integration of spatial data analysis and visualization and also offers user-friendly mechanisms for customized map visualization.

6.2. Florida Lung Cancer Analysis

In this section we provide an example of how our GeoCloud system can be employed in epidemiologic research. Assume a researcher studies lung cancer in Florida. She can upload and choose the mor_price_income dataset to TerraFly GeoCloud - shown in Figure 25. mor_price_income dataset contains median house price, median income, lung cancer mortality, geometry information and name of each county in Florida.

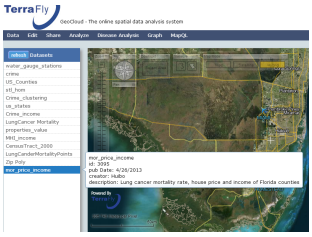


Fig. 25: Datasets in TerraFly GeoCloud

She can then choose the disease analysis button to draw a disease map. In this function, she can choose a legend group number; a disease map is displayed then, as shown in Figure 26.

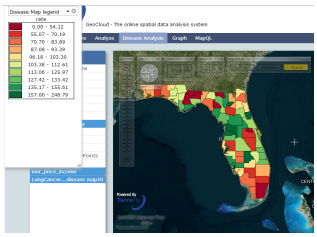


Fig. 26: Lung Cancer disease map

From Figure 26 we observe how this map, with legend at the top left corner, provides a direct summary of the disease data. For lung cancer in Florida, the mortality in the central region is higher and it is lower in the south region. However, the researcher cannot have an accurate analysis result just from this one map. She can further choose the cluster and outlier detection function, which uses Local Morans I to perform further analysis. This analysis function provides three maps: local Morans I map, z-value map, and p-value map. Figure 27 shows the p-value map, from which the researcher can know which counties form a statistically significant cluster and which counties are statistically significant outliers.

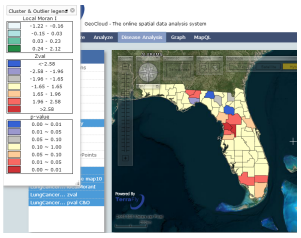


Fig. 27: P-value map of Local Moran I

Now the researcher may want to know what kind of relationship exists between lung cancer mortality and the median income of each county. For this purpose, she can use the median income dataset provided by the TerraFly GeoCloud system, and apply the spatial auto-regression tool. Figure 28 shows the result of this model. From the result, we can observe that when the mortality of surrounding areas increase by 1, the mortality of this county will increase by 0.233, and when the median income in the surrounding area increases by $\$10,000$, the mortality of this county will decrease by 0.09.

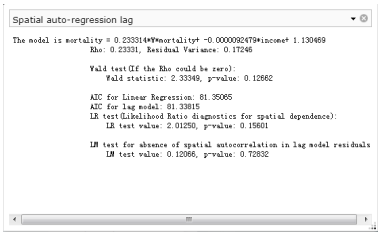


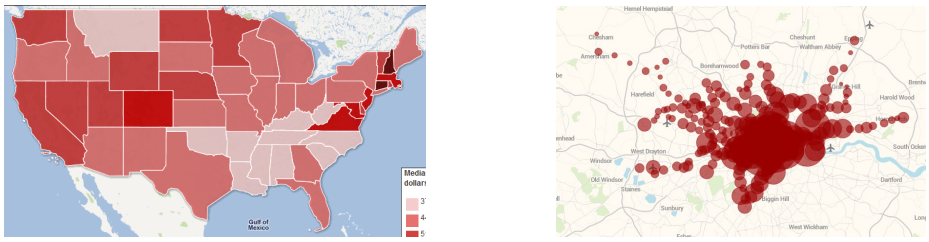
Fig. 28: Spatial auto-regression of lung cancer mortality and median income

7. RELATED WORK AND PRODUCTS

7.1. Spatial Data Visualization

Information visualization (or data visualization) techniques are able to present the data and patterns in a visual form that is intuitive and easily comprehendible, allow users to derive insights from the data, and support user interactions [Zhang and Li 2012; Spence and Press 2000; Li et al. 2010b]. For example, Figure 29a shows the map of Native American population statistics which has the geographic spatial dimensions and several data dimensions. The figure displays both the total population and the population density on a map, and users can easily gain some insights on the data by a glance [Old 2002]. In addition,visualizing spatial data can also help end users interpret and understand spatial data mining results. They can get a better understanding on the discovered patterns.k

Visualizing the objects in geo-spatial data is as important as the data itself. The visualization task becomes more challenging as both the data dimensionality and richness in the object representation increase. In TerraFly GeoCloud, we have devoted lots of effort to address the visualization challenge including the visualization of multi-dimensional data and the flexible user interaction. For spatial data mining to be effective, it is important to include the visualization techniques in the mining process and to generate the discovered patterns for a more comprehensive visual view [Zhang and Li 2012; Rishe et al. 2004].



(a) America Median Income (b) Customized map

Fig. 29: Related work

7.2. Spatial Analysis

Spatial analysis is especially used on geographic data. The difference between spatial analysis and traditional analysis is that spatial analysis methods use spatial information of the data, such as the location, orientation, and adjacent areas. Spatial analysis is widely used in many domains including biology, ecology, epidemiology, ecology, and criminology. There are many kinds of spatial analysis methods which include spatial clustering, spatial autocorrelation, spatial regression, spatial interpolation and spatial distribution measurement [Fotheringham and Rogerson 2013]. TerraFly GeoCloud presents comprehensive spatial analysis methods and result visualization in a more interactive way. User can leverage these methods without programming, and obtain the result visualized on the map with just a few clicks [Bailey et al. 1994].

7.3. Customized Map Visualization

The process of rendering a map generally means taking raw geospatial data and making a visual map from it. Often it applies more specifically to the production of a raster image, or a set of raster tiles, but it can refer to the production of map outputs in vector-based formats. ”3D rendering” is also possible when taking the map data as an input. The ability of rendering maps in new and interesting styles, or highlighting features of special interest, is one of the most exciting aspects in spatial data analysis and visualization.

Customized map visualization have several challenges. First, it takes time to generate a map. User needs to use complicated programs to generate maps from traditional map visualization software tools. Second, it is hard to obtain a really customized map. Some map services can provide some customized views for users. For example, Figure 29b shows a customized map where the adjacent data objects are merged together and are represented using big circles. However, it can not allow users to manipulate the data as there are only few visualization styles are provided.

TerraFly map render engine is a toolkit for rendering maps and is used to render the main map layers. It supports a variety of geospatial data formats and provides flexible styling options for designing many different kinds of maps, and the render speed is fast [Teng et al. 2006; Lu et al. 2014]. TerraFly Geocloud also provides MapQL as a spatial query and map render tool. User can query and visualize the data use a SQL-like statements. Because Geocloud is a web-based online service, user can use MapQL online and get a result in the map directly. This SQL-like statements facilitate users and let them draw the map in their own ways [Lu et al. 2013a].

7.4. Related Products

In the geospatial discipline, web-based GIS services can significantly reduce the data volume and required computing resources at the end-user side [Li et al. 2010a; Fotheringham and Rogerson 2013]. To the best of our knowledge, TerraFly GeoCloud is one of the first systems to study the integration of online visualization of spatial data, data analysis modules and visualization customization language.

Various GIS analysis tools are developed and visualization customization languages have been studied in the literature. ArcGIS is a complete, cloud-based, collaborative content management system for working with geographic information. But systems like ArcGIS and Geoda focus on the content management and share, not online analysis [Johnston et al. 2001; Anselin et al. 2006]. Azavea has many functions such as optimal Location find, Crime analysis, data aggregation and visualization. It is good at visualization, but has very limited analysis functions [Boyer et al. 2011].

Various types of solutions have been studied in the literature to address the problem of visualization of spatial analysis. However, on one hand, good analysis visualization tools like Geoda and ArcGIS do not have online functions. To use them, users have to download and install the software tools, and download the datasets. On the other hand, good online GIS systems like Azavea, SKE, and GISCloud have limited analysis functions. Furthermore, none of above products provides a simple and convenient way like MapQL to let user create their own map visualization [Hearnshaw et al. 1994; Boyer 2010]. The related products are summarized in Table I. Our work is complementary to the existing works and our system also integrates data mining and visualization.

Table I: GIS Analysis & Visualization Products

Name	Website	Product features description	Online tool	Spatial analysis abilities	Spatial visualization abilities
ArcGIS	http://www.esri.com/ software/arcgis/ arcgis-for-desktop	This software provides map creating and multiple analysis functions. But need training.	No	Multiple analysis functions are provided.	Good visualization. But map creating is complicated and need training.
Geoda	http://geodacenter.asu.edu/	User can import map, add layer to do some geodata analysis.	No	Multiple analysis functions, such as statistic map and rate map.	Limited visualization.
ArcGIS Online	http://www.arcgis.com	ArcGIS Online is a complete, cloud-based, collaborative content management system for working with geographic information.	Yes	No online Analysis.	Focus on the content management and share.
Azavea	http://www.azavea.com/ products/	optimal Location find, Crime analsis, data aggregated and visualized	Yes	Very limited analysis functions	Good visualization.
SKE	http://www.skeinc.com/ GeoPortal.html	Spatial data Viewer	Yes	Very limited simple analysis.	Focus on the spatial data viewer.
GISCloud	http://www.giscloud.com	with few analysis (Buffer , Range , Area , Comparison , Hotspot , Coverage , Spatial Selection)	Yes	No spatial analysis function	Focus on geo-data management and share.
GeoIQ	http://www.geoiq.com/ http://geocommons.com/	filtering, buffers, spatial aggregation and predictive	Yes	Very limited and simple analysis: currently provide predictive (Pearsons Correlation).	Focus on GIS, very good visualization and interactive operation.
GeoCloud	http://terrafly.fiu.edu/GeoCloud/	Provide spatial data visualization, spatial dependency and auto-correlation, spatial data clustering, spatial regression, measuring geographic distribution, spatial interpolation and customize map visualization	Yes	Provides multiple spatial analysis function. Easy to use.	Provide good data visualization and interactive operation. Easy to use.

8. CONCLUSION

This paper presents TerraFly GeoCloud, an online spatial data analysis and visualization system, to facilitate end users to visualize and analyze spatial data, and to share the analysis results. TerraFly GeoCloud focuses on building a new intelligent system that allows a general user perform spatial data analysis in a very simple and convenient way. By leveraging distributed computing, visualization and data mining techniques, TerraFly GeoCloud enables users to perform different types of spatial data analysis tasks for decision support. The system is a GIS analysis tool providing software as a service (SaaS). Comparing with traditional desktop software tools, Terrafly GeoCloud is

based on the cloud architecture and users can upload, visualize, analyze, and share the data through browsers with a few clicks. As the application of cloud service is getting widely used, this type of intelligent systems will be more and more popular in the future. About the future works, we will provide better visualization techniques to improve user experience. As user visits increasing, we will add load balance function in the front end through some popular technologies such as Nejx.

sed in part upon work supported by the National Science Foundation under Grant Nos. I/UCRC IIP-1338922, AIR IIP-1237818, SBIR IIP-1330943, III-Large IIS-1213026, MRI CNS-0821345, MRI CNS-1126619, CREST HRD-0833093, I/UCRC IIP-0829576, MRI CNS-0959985, FRP IIP-1230661, SBIR IIP-1058428, SBIR IIP-1026265, SBIR IIP-1058606, SBIR IIP-1127251, SBIR IIP-1127412, SBIR IIP-1118610, SBIR IIP-1230265, SBIR IIP-1256641. Includes material licensed by TerraFly (http://terrafly.com) and the NSF CAKE Center (http://cake.fiu.edu).

REFERENCES

Luc Anselin. 1995. Local indicators of spatial associationLISA. *Geographical analysis* 27, 2 (1995), 93–115.

Luc Anselin, Ibnu Syabri, and Youngihn Kho. 2006. GeoDa: an introduction to spatial data analysis. *Geographical analysis* 38, 1 (2006), 5–22.

Peter Armitage, Geoffrey Berry, and John Nigel Scott Matthews. 2008. *Statistical methods in medical research*. John Wiley & Sons.

Trevor C Bailey, S Fotheringham, and P Rogerson. 1994. A review of statistical spatial analysis in geographical information systems. *Spatial analysis and GIS* (1994), 13–44.

Michel Bilodeau, Fernand Meyer, Michel Schmitt, and Georges Matheron. 2005. *Space, Structure and Randommess: Contributions in Honor of Georges Matheron in the Field of Geostatistics, Random Sets and Mathematical Morphology*. Springer.

Deborah Boyer. 2010. From internet to iPhone: providing mobile geographic access to Philadelphia's historic photographs and other special collections. *The Reference Librarian* 52, 1-2 (2010), 47–56.

Deborah Boyer, Robert Cheetham, and Mary L Johnson. 2011. Using GIS to Manage Philadelphia's Archival Photographs. *American Archivist* 74, 2 (2011), 652–663.

HJ De Knegt, F Van Langevelde, MB Coughenour, AK Skidmore, WF De Boer, IMA Heitkonig, NM Knox, R Slotow, C Van der Waal, and HHT Prins. 2010. Spatial autocorrelation and the scaling of species-environment relationships. *Ecology* 91, 8 (2010), 2455–2465.

Robin Dubin, R Kelley Pace, and Thomas G Thibodeau. 1999. Spatial autoregression techniques for real estate data. *Journal of Real Estate Literature* 7, 1 (1999), 79–96.

Paul Elliott and Daniel Wartenberg. 2004. Spatial epidemiology: current approaches and future challenges. *Environmental health perspectives* (2004), 998–1006.

Martin Ester, Hans-Peter Kriegel, J Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise.. In *KDD*, Vol. 96. 226–231.

Stewart Fotheringham and Peter Rogerson. 2013. *Spatial analysis and GIS*. CRC Press.

Arthur Getis and J Keith Ord. 1992. The analysis of spatial association by use of distance statistics. *Geographical analysis* 24, 3 (1992), 189–206.

Hilary M Hearnshaw, David John Unwin, and others. 1994. *Visualization in geographical information systems.* John Wiley & Sons Ltd.

Kevin Johnston, Jay M Ver Hoef, Konstantin Krivoruchko, and Neil Lucas. 2001. *Using ArcGIS geostatistical analyst*. Vol. 380. Esri Redlands.

Harry H Kelejian and Ingmar R Prucha. 1998. A generalized spatial two-stage least squares procedure for estimating a spatial autoregressive model with autoregressive disturbances. *The Journal of Real Estate Finance and Economics* 17, 1 (1998), 99–121.

Martin Kulldorff. 1997. A spatial scan statistic. *Communications in Statistics-Theory and methods* 26, 6 (1997), 1481–1496.

Martin Kulldorff and Neville Nagarwalla. 1995. Spatial disease clusters: detection and inference. *Statistics in medicine* 14, 8 (1995), 799–810.

Alvin CK Lai, Tracy L Thatcher, and William W Nazaroff. 2000. Inhalation transfer factors for air pollution health risk assessment. *Journal of the Air & Waste Management Association* 50, 9 (2000), 1688–1699.

Hongfei Li, Catherine A Calder, and Noel Cressie. 2007. Beyond Moran's I: testing for spatial dependence based on the spatial autoregressive model. *Geographical Analysis* 39, 4 (2007), 357–375.

Lei Li, Dingding Wang, Chao Shen, and Tao Li. 2010b. Ontology-enriched multi-document summarization in disaster management. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*. ACM, 819–820.

Xiaoyan Li, Liping Di, Weiguo Han, Peisheng Zhao, and Upendra Dadi. 2010a. Sharing geoscience algorithms in a Web service-oriented environment (GRASS GIS example). *Computers & Geosciences* 36, 8 (2010), 1060–1068.

Yun Lu. 2013. Geospatial Data Indexing Analysis and Visualization via Web Services with Autonomic Resource Management. (2013).

Yun Lu, Mingjin Zhang, Tao Li, Yudong Guang, and Naphtali Rishe. 2013a. Online spatial data analysis and visualization system. In *Proceedings of the ACM SIGKDD Workshop on Interactive Data Exploration and Analytics*. ACM, 71–78.

Yun Lu, Mingjin Zhang, Shonda Witherspoon, Yelena Yesha, Yaacov Yesha, and Naphtali Rishe. 2013b. SksOpen: Efficient Indexing, Querying, and Visualization of Geo-spatial Big Data. In *Machine Learning and Applications (ICMLA), 2013 12th International Conference on*, Vol. 2. IEEE, 495–500.

Yun Lu, Ming Zhao, Lixi Wang, and Naphtali Rishe. 2014. v-TerraFly: large scale distributed spatial data visualization with autonomic resource management. *Journal Of Big Data* 1, 1 (2014), 4.

Nathan Mantel. 1967. The detection of disease clustering and a generalized regression approach. *Cancer research* 27, 2 Part 1 (1967), 209–220.

Nikolaos Matsatsinis and Yannis Siskos. 2003. *Intelligent support systems for marketing decisions*. Vol. 54. Springer.

Patrick AP Moran. 1950. Notes on continuous stochastic phenomena. *Biometrika* 37, 1-2 (1950), 17–23.

L John Old. 2002. Information Cartography: Using GIS for visualizing non-spatial data. In *Proceedings, ESRI International Users' Conference, San Diego, CA*.

Stan Openshaw, Martin Charlton, Colin Wymer, and Alan Craft. 1987. A mark 1 geographical analysis machine for the automated analysis of point data sets. *International Journal of Geographical Information System* 1, 4 (1987), 335–358.

Naphtali Rishe, Shu-Ching Chen, Nagarajan Prabakar, Mark Allen Weiss, Wei Sun, Andriy Selivonenko, and D Davis-Chu. 2001. TERRAFLY: A High-Performance Web-based Digital Library System for Spatial Data Access.. In *ICDE Demo Sessions*. 17–19.

N Rishe, M Gutierrez, A Selivonenko, and S Graham. 2005. TerraFly: A tool for visualizing and dispensing geospatial data. *Imaging Notes* 20, 2 (2005), 22–23.

Naphtali Rishe, Yanli Sun, Maxim Chekmasov, Andriy Selivonenko, and Scott Graham. 2004. System architecture for 3D terrafly online GIS. In *Multimedia Software Engineering, 2004. Proceedings. IEEE Sixth International Symposium on*. IEEE, 273–276.

Robert Spence and A Press. 2000. Information visualization. (2000).

Michael L Stein. 1999. *Interpolation of spatial data: some theory for kriging*. Springer.

William Teng, Naphtali Rishe, and Hualan Rui. 2006. Enhancing access and use of NASA satellite data via TerraFly. In *Proceedings of the ASPRS 2006 Annual Conference*.

Jon Wakefield and Paul Elliott. 1999. Issues in the statistical analysis of small area health data. *Statistics in medicine* 18, 17-18 (1999), 2377–2399.

Huan Wang. 2011. A large-scale dynamic vector and raster data visualization geographic information system based on parallel map tiling. (2011).

Huibo Wang, Yun Lu, Yudong Guang, Erik Edrosa, Mingjin Zhang, Raul Camarca, Yelena Yesha, Tajana Lucic, and Naphtali Rishe. 2013. Epidemiological Data Analysis in TerraFly Geo-Spatial Cloud. In *Machine Learning and Applications (ICMLA), 2013 12th International Conference on*, Vol. 2. IEEE, 485–490.

Yi Zhang and Tao Li. 2012. DClusterE: A Framework for Evaluating and Understanding Document Clustering Using Visualization. *ACM Transactions on Intelligent Systems and Technology (TIST)* 3, 2 (2012), 24.

Weizhong Zhao, Huifang Ma, and Qing He. 2009. Parallel k-means clustering based on mapreduce. In *Cloud Computing*. Springer, 674–679.

Sagit Zolotov, Dafna Ben Yosef, Naphtali D Rishe, Yelena Yesha, and Eddy Karnieli. 2011. Metabolic profiling in personalized medicine: bridging the gap between knowledge and clinical practice in Type 2 diabetes. *Personalized Medicine* 8, 4 (2011), 445–456.

Exploring Anomalies in GAStech
VAST 2014 Mini Challenge 1 and 2

Jaegul Choo*, Yi Han*, Mengdie Hu*, Hannah Kim*, James Nugent*, Francesco Poggi† Haesun Park* John Stasko*
*Georgia Institute of Technology, †University of Bologna

ABSTRACT

We present our process and analysis for VAST 2014 Mini Challenge 1 and 2, which integrate an off-the-shelf tool, Jigsaw, rapid web-based visualization prototyping using D3, and analytics-based visualizations using Matlab.

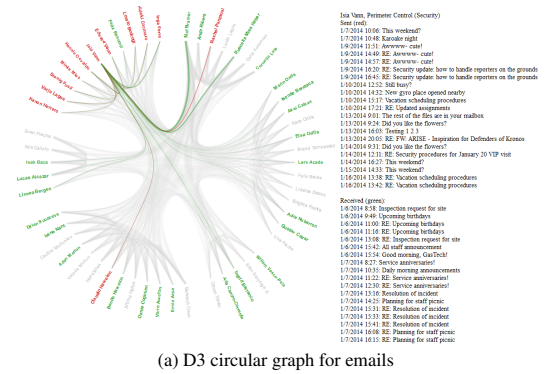
Index Terms: H.5.2 [INFORMATION INTERFACES AND PRESENTATION]: User Interfaces—Theory and methods

1 MINI CHALLENGE 1

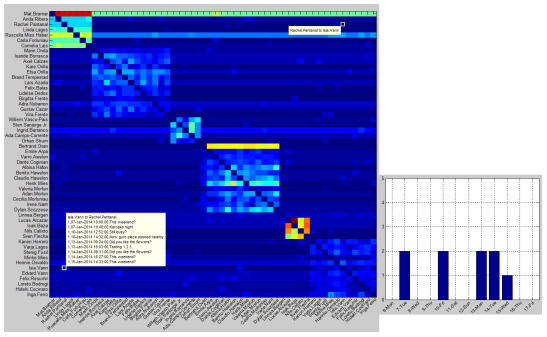
The goals of MC1[1] were to understand the structure of the Protectors of Kronos (POK) organization, the connections of POK to the GAStech company, the series of events that occurred around the time of the challenge's focus incident, and its potential causes. The provided data included semi-structured text documents (news articles, resumes, etc.), structured tabular text documents (email and employee records), an organization chart, and a map.

To examine the news articles and email messages, we used Jigsaw, a visual analytics system for exploring large text document collections [1]. We used Jigsaw's List View to identify a list of news articles around the day of the incident to answer question 2. Subsequently, we deliberately inspected the identified documents in the Document View to find relevant information.

We believed that the email messages between employees would be best visualized with a graph to show the connections. This information was important for answering question 1d about connections between POK and GAStech. We used the Graph View in Jigsaw to identify the different employees involved in a specific email thread. However, the Graph view could not show an overview of the distribution of direct email exchanges between employees. Thus, we used two other visualization tools to explore the overall email connections between any given pair of employees. The first one was a circular graph visualization built using D3 toolkit. As shown in Fig. 1(a), the employees are grouped by their departments around the circumference of the graph. The amount of email between each pair of employees is encoded by the width of the connected line. Mousing over any employee, such as Isia Vann in Fig. 1(a), highlights all emails to and from this person in the circular graph and shows additional information on the side. We used this visualization to find potentially interesting connections between employees sharing last names with known POK members and other GAStech employees. The second tool was built with Matlab and employed an adjacency matrix visualization including interactive features to inspect the communication between any two employees. For example, in Fig. 1(b), Isia Vann, who we suspected to be connected with POK, seemed to be in a relationship with Rachel Pantanal, implying she may also be connected to POK.



(a) D3 circular graph for emails



(b) Matlab matrix view for emails

Figure 1: Email analysis

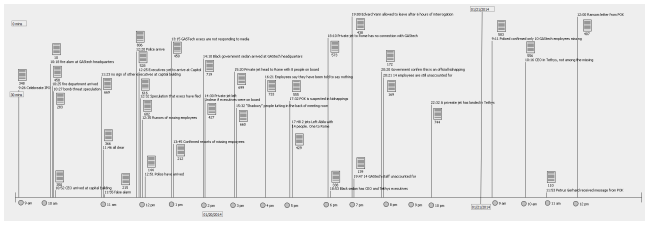


Figure 2: Tablet view for organizing events around the incident

None of these visualizations on its own was sufficient to answer any question in Challenge 1. Instead, we used information from combinations of the tools to gain a better understanding of the data. We did this by using Jigsaw's Tablet window and MS Powerpoint to visually organize our findings from the various visualizations together. The Tablet, shown in Fig. 2, provides a workspace for manually creating visual representations with lines, text boxes, and connections to documents. We created a timeline view in the Tablet for question 2 with notes and links to related documents. The view provides a flexible platform for visually organizing and presenting findings. For information learned outside of Jigsaw, we used MS Powerpoint to gather our findings.

2 MINI CHALLENGE 2

The task of Mini Challenge 2[2] was to find unusual patterns in employees' daily lives from their credit/debit/loyalty card records and

*e-mail: jaegul.choo@cc.gatech.edu, yihan@gatech.edu, mengdie.hu@gatech.edu, hannahkim@gatech.edu, jnugent6@gatech.edu, fpoggi@cs.unibo.it, hpark@cc.gatech.edu, stasko@cc.gatech.edu

[1]Video: http://www.cc.gatech.edu/gvu/ii/challenge/GT-14-MC1.mp4

[2]Video: http://www.cc.gatech.edu/gvu/ii/challenge/GT-14-MC2.mp4

IEEE Symposium on Visual Analytics Science and Technology 2014
November 9-14, Paris, France
978-1-4799-6227-3/14/$31.00 ©2014 IEEE

(a) Timeline view for GAStech

(b) Timeline view for employees' homes

(c) Matrix view for spending data

(d) Timeline view

(e) Space-time cube view

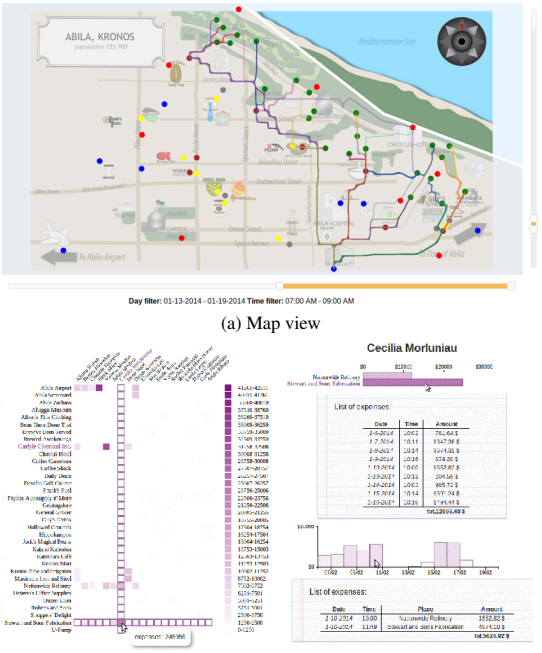
Figure 3: Analytics-based tool

the GPS tracking records of their cars. Given incomplete, inaccurate data, we first identified the exact store locations and the complete car-employee assignments. We then generated each employee's trajectory combined with his/her spending information.

Analytics-based tools. The identified information allowed us to perform analysis based on both particular locations and particular persons. For every location, we created a timeline view visualizing who stayed at the location over time, along with spending records (in the case of shops). Fig. 3(a) shows a timeline view for 'GAStech'. In this view, each row corresponds to a particular employee's timeline, and the time duration of his/her stay is represented as a rectangle along the horizontal timeline, color-coded by his/her department. This figure clearly shows regular business hours for most employees as well as Lucas Alcazar's anomalous pattern of often coming back to the office at night (red circles). Fig. 3(b) shows the timeline views of two places for employees' houses. One can see that Hennie Osvaldo sometimes stays at someone else's place at night. For particular employees, we provided the space-time cube view (Fig. 3(e)) showing their GPS trajectories in 3D where the x-, y-, and z-axes represent longitude, latitude, and time, respectively. Additionally, the matrix view shows the spending amount as a shop-by-day matrix for a particular employee and an employee-by-day one for a particular shop. For example, Fig. 3(c) indicates Isak Baza's regular visits to 'Ouzeri Elian'.

We provided flexible interactions among all the three views. For example, from the timeline view of 'Frydos Autosupply n' More' (Fig. 3(d)), where circles represent spending records with the radius proportional to the amount, we found an unusual spending of $10,000 by Lucas Alcazar. Upon selecting it, the space-time cube that involves those employees visiting this place at the same time pops up (Fig. 3(e)). This interaction revealed that when Lucas' credit card was used at this place, he was not there while Minke Mies was. Furthermore, we can also see another transaction occurring in this manner at 'U-Pump', implying that Minke might have stolen Lucas' credit card.

Web-based tools. For further analysis, we developed two web-based visualization tools. The first[3] provides a zoomable map of Abila, two sliders to filter days and hours, and a combo box to select vehicles. Circles represent shops, whose color indicates shop types (e.g., brown for cafe), and lines represent the vehicle trajectories. We used this tool both to analyze recurring employee's routines, e.g., Fig. 4(a)), showing the routes of the employee commuting to GAStech in the morning during the weekdays) and to further inspect detailed unusual car movements or spending patterns. The second web tool[4] is composed of an interactive heatmap that provides a spending summary and two coordinated bar graphs with details on each expense organized/filtered by employee, shop, and



(a) Map view

(b) Spending view

Figure 4: Web-based tool

date (see Fig. 4(b)).

3 Future Work

We plan to improve the analytics-based tools so that the basic interactions such as brushing-and-linking and details-on-demand types can be easily supported via native command-line interfaces of statistical analytics tools [2].

Acknowledgments

This work was supported in part by NSF grants CCF-0808863 and IIS-0915788, the DARPA XDATA grant FA8750-12-2-0309, and the DHS VACCINE Center of Excellence.

References

[1] C. Görg, Z. Liu, J. Kihm, J. Choo, H. Park, and J. T. Stasko. Combining computational analyses and interactive visualization for document exploration and sensemaking in jigsaw. *IEEE Trans. on Visualization and Computer Graphics*, 19(10):1646–1663, 2013.

[2] C. Lee, J. Choo, D. H. P. Chau, and H. Park. Augmenting matlab with semantic objects for an interactive visual environment. In *Proc. the SIGKDD Workshop on Interactive Data Exploration and Analytics*, pages 63–70, 2013.

[3] http://eelst.cs.unibo.it/vast/map/

[4] http://eelst.cs.unibo.it/vast/heatmap/

Exploring Spatio-Temporal Data as Personal Routes

Alex Godwin, Anand Sainath, Sanjay Obla Jayakumar, Vivek Nabhi, Sagar Raut, John Stasko

Georgia Institute of Technology, Atlanta, Georgia, United States

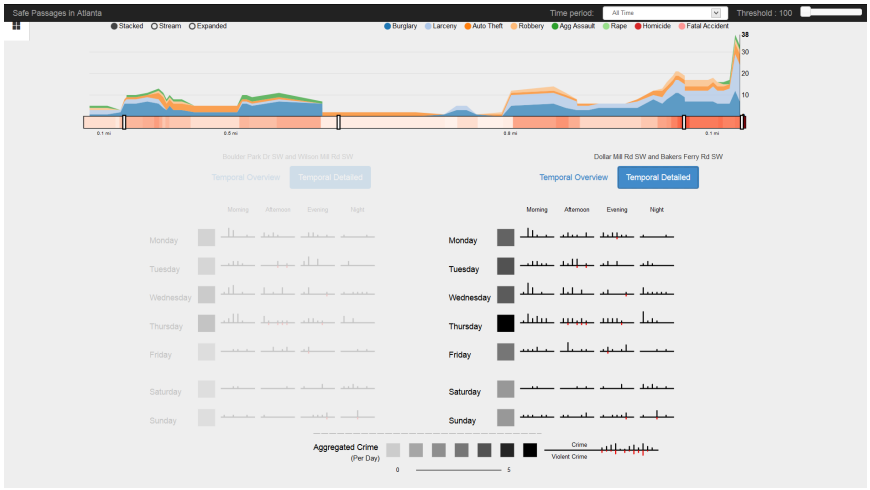


Figure 1: A detailed view of spatio-temporal data along a trajectory.

Abstract

Spatio-temporal data is often displayed using regional aggregation or heatmaps, which are useful for exploring large distributed trends or working to unearth the cause of more localized behavior. For individual users that live and work in the region, however, these representations are inaccessible and difficult to put into practice. We present a new technique for exploring spatio-temporal data as personal routes through a geographic area. With this technique, users are able to examine the details of a subset of event records that are contextually relevant to a trip taken through the area of consideration. Our technique can be applied to any spatio-temporal data that consists of point events, and is demonstrated through a visualization system, Safe Passage, that displays crime data from an urban area in the context of pedestrian routes that users take through the city.

Keywords: Spatio-temporal analysis, timelines, non-expert visualization, personal routes, crime data

1 Introduction

Geospatial analysis tools provide significant benefits to urban planners and civic figures responsible for shaping the geospatial distribution of an area. Such tools help planners to understand correlations between events in the region (e.g., crime) and regional factors that influence those events. The same geospatial analysis tools do not present a compelling use case for individuals, however, in which the information presented can be made actionable. For instance, a person's daily commute offers one opportunity for translating spatial event data into decisions.

* alex.godwin@gatech.edu

This poster introduces a new technique for visualizing event data by moving the focus to the specific routes that a user can take through a city. Our technique displays event information along a route, or series of waypoints, with contextual information about the types of events that occur along that route. This visualization allows people to construct personal routes based upon their starting location and destination, and update that route as new event trends emerge. We provide an interface in which a person enters a route (e.g., from home to work) and reviews detailed information about that route through a combination of maps and temporal views. This technique is demonstrated through analysis of Atlanta metro crime data.

2 Related Work

Personal route visualization is a natural extension of previous work in two areas of research: geospatial visualization of event data and representation and interaction with trajectory data that reflects a course through a geospatial region. Event data, consisting of a fixed location in space and time, is often represented through the use of maps. Many cartographic representations aggregate events into fixed regions, which are limited to showing a single time slice only. Crime hotpot analysis can go further to reveal characteristics of the physical locations and times of day that contribute to higher crime rates.

Many representations of trajectories emphasize analysis of a large pool of trajectories rather than focused exploration of an individual trajectory. By aggregating a large database of trajectories together, a user can begin to identify temporal trends as well as spatial trends [1, 2]. In these systems, the trajectories themselves are the objects of interest rather than the contextual data surrounding the trajectories. An alternative approach is to focus on data—unrelated to the velocity or orientation of the trajectory—such as a variable measured along its path. In transect sampling, for example, animal sightings along a tour taken through a wildlife space have long been used to estimate the prevalence of that animal population within a broader region [3].

Visualization techniques have been proposed for representing similar tours through a space in which data, such as radiation along the Tokyo-Fukushima highway, is collected by sampling along a path or computed based upon changes in the path over time [4]. These approaches, however, are still based on an approach in which movement data is used to infer generalizable principles about the region, rather than using a path as a query for exploring known data about a region.

3 Personal Route Visualization

First, we compose a transect that includes only the relevant spatial events within the region. We then construct a trajectory between two endpoints consisting of an origin and a destination. Then, the trajectory is subdivided into constituent spatial event waypoints at regular intervals. A user-specified sampling distance is then used to sample a subset of the spatial event data points from each waypoint location along the trajectory, reducing the number of events under consideration to only those that are near the trajectory. The sampled spatial events are then associated with the waypoints that they occur in proximity to. Once the associations between spatial events and waypoints have been made, the transect can then be characterized by these associations in space and time.

3.1 Route View

We construct a personal route that can be used to navigate from one location to another. A subset of optimal trajectories are generated using a directions service (e.g., Google Maps API), that conform to the constraints provided by the user. A map-based representation is displayed of the routes (Figure 2). The spatial event data is sampled at major waypoint milestones (turns and street changes). The result is a node-link diagram in which the nodes represent the waypoints and the links are used to connect nodes that are located within the same route. The area of each node along the route is used to indicate the number of spatial events that occur within the sampled area of that waypoint. The location of each node roughly conforms to the original position of the waypoint on the map, but has been perturbed by a force-directed layout to avoid over-plotting of the node ellipses.

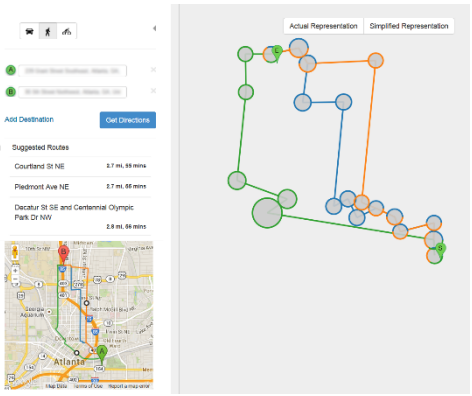


Figure 2: Route view showing both a map-based display of the true route and a simplified node-link view of the crime that occurs at each of the major waypoints.

3.2 Timeline View

A timeline view displays the changes in spatial event prevalence over the course of the selected route (Figure 1). The default and starting layout of the timeline is stacked area chart in which the x-axis represents the distance from the origin and the y-axis represents the amount of event occurrence. Along the bottom of the stacked area chart, a secondary visualization has been added in which vertical markers are placed along the axis in the corresponding location for each of the major waypoints along the route (e.g., turn locations). These markers are embedded in a reinforcing heatmap in which the saturation is higher in regions where the spatial events are more prevalent. Hovering the mouse over a waypoint markers provides a tooltip for the turn guidance provided by the directions service API (e.g., "Turn Left onto Main St SE").

3.3 Detailed Temporal View

While the timeline along the top of the screen is useful for understanding the historical context of all spatial events that have occurred along the transect of the route, the types of event present can vary significantly depending on when you depart. A temporal view is generated for each of the alternate routes displayed in the route view. These temporal views are arranged along the bottom of the screen (Figure 1). The currently selected route is emphasized by reducing the opacity of the other temporal views. In these temporal views, the times of day are displayed as timelines that depict the relative prevalence of events depending on when the user takes the trip. Bars indicate the prevalence of events in general through height, while a red-shaded bar underneath each axis reflects the prevalence of incidents of high interest (e.g., violent crime prevalence).

4 Future Work

There are two primary areas of future work that could extend the techniques described in this paper. First, the distinction between expert and non-expert tasks could be further examined with respect to the types of tasks that personal routes could support for experts. One method for accomplishing this would be to modify the existing transect construction technique so that the distance from the spatial event to either the waypoint or trajectory are taken into account to create a more nuanced model than the current sampled count of the proximal spatial events. Second, our technique could be applied to additional data sets to better establish the domains in which it provides value. We have begun to explore automobile accident records within the context of personal routes, and have seen some initial success in displaying the areas along a trajectory in which accidents occur more frequently.

5 Acknowledgements

The authors wish to the Atlanta Police Department for providing the open data sources used in the case studies. Support for the research provided by the National Science Foundation (CCF-0808863) and the DHS VACCINE Center of Excellence.

References

[1] Gennady Andrienko and Natalia Andrienko, "Spatio-temporal aggregation for visual analysis of movements," in *Visual Analytics Science and Technology (VAST '08)*, Columbus, Ohio, 2008, pp. 51-58.

[2] Gennady Andrienko, Natalia Andrienko, and Stefan Wrobel, "Visual Analytics tools for analysis of movement data," *ACM SIGKDD Explorations Newsletter*, vol. 9, no. 2, pp. 38-46, 2007.

[3] Stephen T Buckland, David R Anderson, Kenneth P Burnham, and Jeffrey L Laake, *Distance Sampling*.: John Wiley & Sons, 2005.

[4] Christian Tominski, Heidrun Schumann, Gennady Andrienko, and Natalia Andrienko, "Stacking-based visualization of trajectory attribute data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2565-2574, 2012.

Information Visualization
http://ivi.sagepub.com/

Reflections on the evolution of the Jigsaw visual analytics system

Carsten Görg, Zhicheng Liu and John Stasko

Information Visualization 2014 13: 336 originally published online 23 July 2013
DOI: 10.1177/1473871613495674

The online version of this article can be found at:
http://ivi.sagepub.com/content/13/4/336

Published by:
SAGE
http://www.sagepublications.com

Info Vis

Article

Information Visualization
2014, Vol. 13(4) 336–345

DOI: 10.1177/1473871613495674
ivi.sagepub.com
SAGE

Reflections on the evolution of the Jigsaw visual analytics system

Carsten Görg[1], Zhicheng Liu[2] and John Stasko[3]

Abstract

Analyzing and understanding collections of textual documents is an important task for professional analysts and a common everyday scenario for nonprofessionals. We have developed the Jigsaw visual analytics system to support these types of sensemaking activities. Jigsaw's development benefited significantly from the existence of the VAST Contest/Challenge that provided (1) diverse document collections to use as examples, (2) controlled exercises with a set of analytic tasks and solutions for judging results, and (3) visibility and publicity to help communicate our ideas to others. This article describes our participation in a series of VAST Contest/Challenge efforts and how this participation helped influence Jigsaw's design and development. We describe how the system's capabilities have evolved over time, and we identify the particular lessons that we learned by participating in the challenges.

Keywords

VAST Contest, visual analytics, investigative analysis, intelligence analysis, information visualization, sensemaking, multiple views

Introduction

Suppose that you are given a big box full of the pieces from many different jigsaw puzzles and you are asked to put the pieces together from one or two of the most "interesting" puzzles and describe what you see. Oh, by the way, not all the pieces of those "interesting" puzzles are in the box. Investigative analysts, particularly those in fields such as law enforcement or intelligence, frequently confront this kind of challenge in their work. They are given large collections of seemingly unconnected documents and are tasked with identifying a plot or threat that is hinted at, but not clearly communicated, by a small subset of the documents in the collection.

We have developed a visual analytics system called Jigsaw[1,2] to help investigators faced with such challenges. Jigsaw provides a suite of visualizations that depict different perspectives on the documents and the entities (people, places, organizations, etc.) within these documents. Each visualization (called a "view" in Jigsaw) communicates a different aspect of the documents and how the different entities relate to each other. Jigsaw allows an analyst to search for a particular entity and then the system visually communicates the context of that entity, such as the documents in which it appears and the other entities to which it is connected. Alternately, Jigsaw provides different overviews of the document collection so that an analyst can gain some initial evidence about where to begin exploring in more depth. Jigsaw does not automatically find suspicious threads throughout the collection or tell an analyst what to examine first. Instead, it acts

[1]Computational Bioscience Program, School of Medicine, University of Colorado, Aurora, CO, USA
[2]Department of Computer Science, Stanford University, Stanford, CA, USA
[3]School of Interactive Computing, Georgia Institute of Technology, Atlanta, GA, USA

Corresponding author:
Carsten Görg, Mail Stop 8303, 12801 E 17th Ave, Aurora, CO 80045, USA.
Email: Carsten.Goerg@ucdenver.edu

more as a visual index, helping to show which documents are connected to each other and which are relevant to a particular line of investigation being pursued.

History and contest/challenge participation

In 2004 and 2005, John Stasko participated in a series of meetings that helped to develop an initial definition and research agenda for the field of visual analytics, documented in the book *Illuminating the path*.[3] At those meetings, Professor Frank Hughes of the Joint Military Intelligence College conducted two analysis exercises to provide attendees with a good example of the type of work investigators often conduct. Each exercise involved a set of short, synthetic intelligence reports. These reports were one to a few paragraphs in length and described events that would be of interest to law enforcement and intelligence officials. The events included specific details about particular people, places, organizations, and dates. Attendees were tasked with assimilating these reports, making sense of their contents, and most crucially "connecting the dots" to synthesize a larger crime or threat in the planning stages. No one document itself was enough to understand the plot. A variety of information had to be integrated from across the documents to construct a more complete narrative. These exercises were done using pencil and paper, which are close to representing the state of tools used by many analysts in the field at that point.

Working on these exercises was very challenging for the workshop attendees, even when the exercises consisted of relatively few documents. In particular, it was difficult to keep all the relevant entities clear in one's mind, to remember the context at which they were discussed, and to connect them to other activities that were noted. It occurred to us that visual analytics might provide capabilities to assist with such investigative, sensemaking, and knowledge synthesis tasks. This realization was the genesis of Jigsaw, and it motivated us to create a system that would help investigators and analysts who are confronted with large document collections and need to rapidly understand their contents.

We began the design of Jigsaw in 2006 and shortly thereafter built the initial visualizations within the system. The VAST Contest started in 2006, but our system was not mature enough to be used at that time. In early 2007, the second annual VAST Contest was released. It consisted of approximately 1500 news reports (short text documents) each of a few paragraphs. We decided that this collection would be a good test bed for our system.

We started working on the problem by dividing the news report collection into four piles (for the four people on our team doing the investigation). Each of us skimmed the 350+ reports in our own unique subset just to become familiar with general themes discussed in those documents. We also jotted down notes about people, organizations, or events to potentially study further.

Next, we came together to examine the entire news report collection. Using Jigsaw, we explored a number of the potential leads that each person identified in the initial skim of the reports. At first, we looked for connections across entities, essentially the same people, organizations, or incidents being discussed in multiple reports. After about 6 hours of exploration, we really had no definite leads and were left with many possibilities. Therefore, we returned to the text reports, and some team members read subsets of the reports they had not examined before. At that point, we identified some potential interesting activities and themes to examine further. What also became clear was that the time we spent earlier exploring the documents in Jigsaw was not wasted. It helped us become more familiar with many different activities occurring in the reports. Closer, more deliberate examinations and readings of the documents uncovered more promising leads and we found additional connections across some actors and organizations in the dataset. Ultimately, we discovered a sinister plot in which animals smuggled into the country were infected with a serious disease that could be transmitted to humans. The contest judges viewed our entry as being extremely accurate about the potential threat, and we were declared the top entry within the academic division of the contest that year. Details of our Contest entry and the analytical process are discussed in Görg et al.[4,5]

Some years later, we entered the VAST Challenge (as it was now known) in 2010. In particular, Mini Challenge 1 in 2010 provided a document collection and an objective much like in 2007—identify a latent threat across the collection and describe the particular details involved in that threat. Unlike the larger collection in the VAST Contest of 2007 where one had to find the "needle in the haystack," here, many different documents contributed to a complex, multifaceted storyline. The plot involved arms dealers from different countries who all convened at a particular location.

Since the number of documents in the dataset was relatively small (just over a hundred compared to more than a thousand in the 2007 Contest), we were able to quickly familiarize ourselves with most of the documents using the views in Jigsaw. We soon realized that unlike in the 2007 Contest data where only a small subset of the documents was relevant to the final solution, most documents in this Mini Challenge seemed to contribute to a larger story.

We used some of Jigsaw's new functionality—computational text analyses and evidence

marshaling—to scaffold our investigation. We started our exploration by examining the high-frequency entities and their connections in the List View and the Graph View. This enabled us to directly focus our attention on important people and places in the dataset. Showing document clusters grouped by topics in the Document Cluster View helped us to keep track of the different threads of the stories embedded in the dataset; in addition, this view indicated which documents we had already read and explored. We created multiple pages in the Tablet (our new approach for note-taking). The pages organized our findings and thinking processes in terms of different perspectives and themes, including social networks, timelines, specific topics such as weapon and fund transfers, and geographically connected people and events. We iteratively modified and refined our hypotheses and findings represented in the Tablet as we read the documents in greater depth and discovered connections between interesting entities. In the end, we uncovered a social network illustrating associations among key players in the arms dealing, patterns of people meeting arms dealer Nicolai Kuryakin in Dubai in the period between April 17 and April 20 in 2009, and patterns of bank fund transfers. Our Challenge entry won an award for "Good Support for Data Ingest." Details of our entry and the analytical process we used for the investigation are discussed in Liu et al.[6]

The 2011 VAST Challenge and its Mini Challenge 3 provided a much larger document collection to explore. It contained 4744 text documents, each in the form of a news report. Jigsaw's organizational and filtering capabilities helped narrow the collection and made it possible for us to browse many of the documents rapidly. As we explored and read more documents, we began to notice that the majority of the documents in the collection were modified versions of actual news articles from the 1990s with key entity names changed. Ultimately, we believed that these documents were not related to the embedded challenge plot. Other interesting and potentially relevant documents, however, were typically shorter and seemed to center around recent activities at a fictitious city called Vastopolis. We uncovered organizations that were planning to make a bioterrorist attack on the city. For this Mini Challenge, Jigsaw was most useful for rapid triage on the documents, helping to determine their potential relevance to the plot. It provided multiple analytical perspectives on the documents' text. Our Challenge entry won an award for "Good Use of the Analytic Process." Details of our entry and the analytical process are discussed in Braunstein et al.[7]

In the following sections, we describe how our participation in the VAST Contest/Challenge influenced the evolution of the Jigsaw system (and our design decisions in particular), and we describe the lessons we learned. We have covered related work of visualization and visual analytics approaches for textual data in two previous journal articles[1,2] and therefore do not provide an explicit section on related work in this article. We do discuss work of other researchers who used Jigsaw to work on their own data in section "Adoption and dissemination."

Lessons learned and Jigsaw evolution

When we started to work on the VAST '07 Contest, we had just finished the implementation of the first prototype of the Jigsaw system. Grounded in our expertise as visualization researchers, the system heavily relied on the interactive visual representation of connections between entities identified across textual documents, and it did not provide any kind of automated text analyses, such as document clustering or summarization. It neither supported the automated identification of named entities, such as people, places, and organizations, in the documents, so we solely relied on the provided dataset, which included identified entities. Details about the state of the system at that point are described in a previous article.[8] In this section, we discuss how our experience from participating in the VAST Contest/Challenges influenced our design decisions and the development of the Jigsaw system.

Reading the documents still matters

One important lesson we learned from working on the contest datasets is that the interactive visualization of connections between entities and documents alone cannot replace the reading of reports. Repeatedly and carefully reading reports is crucial to incrementally expand knowledge about the dataset and to understand details in the underlying plot. The initial version of Jigsaw was helpful in this respect by identifying a small subset of reports that are relevant to an idea being explored and that can be examined closely. However, besides a basic Text View, no other view was tailored towards the visualization of textual data.

To address this shortcoming, we integrated automated text analysis, such as text summarization and clustering, into later versions of the system and made them available throughout the views. These analyses can facilitate the reading of documents. We also improved the Text View (renamed as the Document View) itself since it is such an important component of the system. The views in Figure 1(a) and (b) show the four documents mentioning *Luella Vedric*. The initial Text View (Figure 1(a)) only displayed a document with highlighted entities. The documents in the currently loaded document set were represented as tabs,

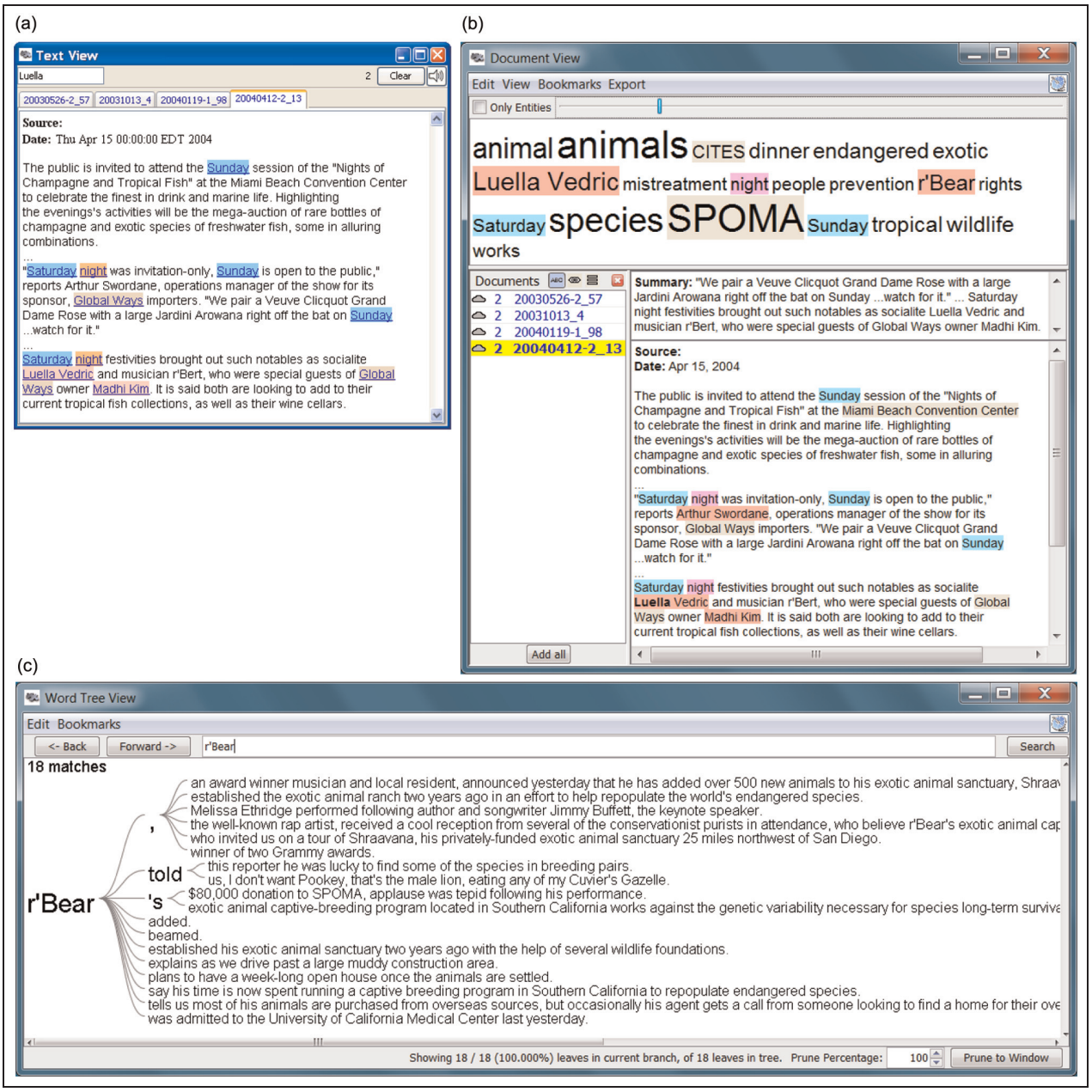


Figure 1. (a and b) The evolution of the Document View. (a) The initial Text View and (b) the current Document View with tag cloud and one-sentence summary of the displayed document. Both views show the same set of documents mentioning *Luella Vedric*. (c) The Word Tree View for *r'Bear*, summarizing the 18 sentences that mention him across 1500 documents.

limiting the number of documents in the view to a few dozen. The current version of the Document View (Figure 1(b)) provides more functionality. It stores the set of loaded documents in a scrollable list (left panel) and thus can handle thousands of documents. It displays a tag cloud (top) for all the documents in the current set to summarize their content.

In this example, we see that the person *r'Bear* in the Blue Iguanodon dataset (2009) and the organization *SPOMA* are key entities in the documents mentioning *Luella Vedric*. The selected document in the set of documents (yellow background) is displayed with highlighted entities (right). Affiliated entities that are connected to a document but are not mentioned in it (e.g. document metadata) are displayed below the document text (not shown in this view because of space constraints); a one-sentence summary of the document is displayed above the document to facilitate the quick scanning of documents. We modified the Document View to count the number of times a

document had been viewed and to allow each view to be named. We frequently found our investigations to have many Document Views present, each with a small number of reports, and naming the view allowed us to recall what the focus of the view was.

To support reading across documents, we implemented Wattenberg's and Viégas' Word Tree approach.[9] The Word Tree View shows all occurrences of a word or phrase across all documents in the context of the words that follow it, each of which can be explored further by a click. The Word Tree View in Figure 1(c) shows occurrences of the person *r'Bear* and the most common phrases that follow that word in sentences within the documents of the Blue Iguanodon dataset (2009). The view illustrates that besides being a musician, *r'Bear* is also involved with the *SPOMA* organization and funds an exotic animal sanctuary, important details in the plot.

Flexible data import is challenging, but vital

Importing and processing data from various sources and in different formats is a crucial feature of any visual analytics system that evolves beyond a lab prototype. Even though we focused our research efforts on the visualization aspects and the integration of computational analyses, we also spent a considerable amount of time and effort on features to ingest and process data. For our participation in the VAST '10 Challenge, we integrated a number of packages to automatically identify entities in text documents, including GATE,[10] LingPipe (http://alias-i.com/lingpipe), the OpenCalais web service (http://www.opencalais.com), and the Illinois Named Entity Tagger.[11] Additionally, we implemented a rule-based approach where we define regular expressions that match dates, phone numbers, zip codes, as well as email, web, and Internet Protocol (IP) addresses. Finally, we added a dictionary-based approach that allows analysts to provide dictionaries for domain-specific entity types that are identified in the documents using basic string matching. All these approaches can be applied to plain text documents, PDF documents, Word documents, and HTML documents. We also implemented a reader for CSV and Excel files that can extract a column with textual data and link it with attributes in other columns.

The improved data import functionality turned out to be very useful beyond our Challenge submission. It is a powerful feature for Jigsaw users outside of academia who have to work with all kinds of text files on a daily basis. Jigsaw users in the law enforcement domain found the reader for Excel files especially useful, and journalists working with Jigsaw often imported their documents from PDF files. These external users discovered a number of bugs and issues in our import functionality and helped us further improve Jigsaw. However, it also became clear to us that we were not able to address all types of exceptions that exist in various file formats and that the data import functionality of a research prototype—which is an important part of a system but not a research contribution—will never be as complete as the data import functionality of a commercial product.

Finally, we defined an XML-based data file format (.jig file) that describes the attributes of documents and the entities within them. In addition to importing files in that format, Jigsaw can also generate these files for documents that were originally imported from other file formats, for example, Word files. This supports the easy sharing of datafiles among Jigsaw users. We have made a number of datasets available in the Jigsaw file format (http://www.cc.gatech.edu/gvu/ii/jigsaw/datafiles.html).

Entity identification is imperfect and needs help

Although algorithms and libraries for entity identification have improved significantly, they are still far from perfect. In this context, we found during our investigation of the VAST Contest dataset that the missing functionality of being able to change the identified entities on the fly was a significant drawback. Since Jigsaw uses the co-occurrence of entities to build a connection network, it is crucial that the entities are properly identified. Missing or unidentified entities result in a knowledge gap: connections that are not there cannot be visualized. Thus, we added a feature to address this issue. Through direct manipulation in its Document View, Jigsaw now supports manually adding entities that were missed by the entity identification process, changing the type of, or altogether deleting wrongly identified entities. Additionally, we addressed the aliasing or duplication problem: the same logical entity may be identified by different strings in different documents. Jigsaw now provides an operation that allows analysts to merge different entities (strings) under one alias. After assigning a primary identifier to the merged entities, that identifier represents all the initially different entities in Jigsaw's visualizations. Jigsaw uses italics to indicate entities with aliases. The alphabetic sort function in the List View can be helpful to find similar entities.

Assist analysts to start an investigation

At the beginning of an investigation, the amount of data to consider is often overwhelming, and it is difficult for analysts to find a starting point. This is especially true for open-ended, strategic analysis scenarios

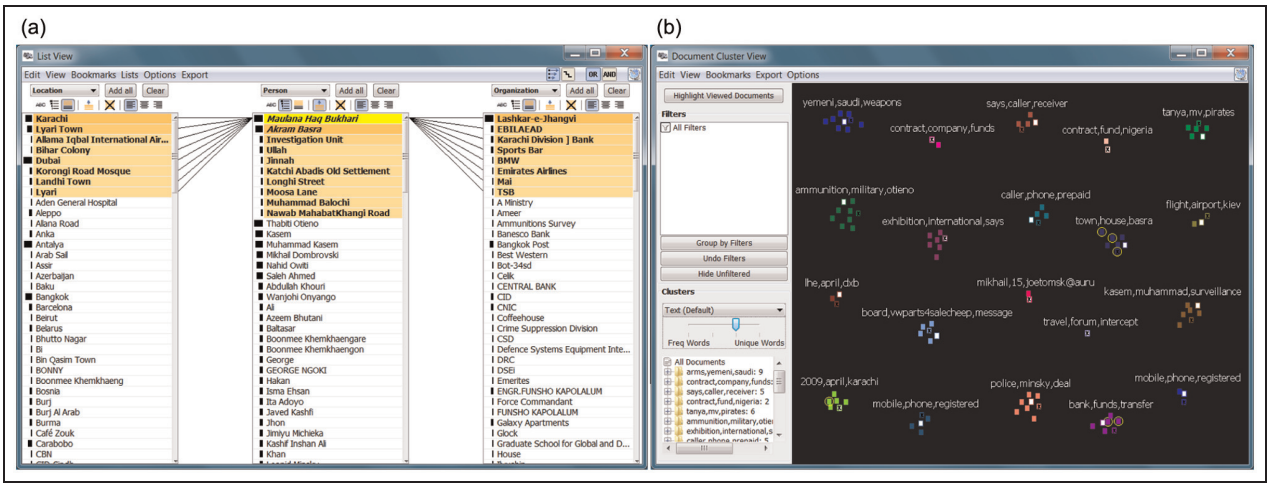


Figure 2. (a) List View showing locations, persons, and organizations connected to *Maulana Haq Bukhari* and (b) Document Cluster View showing different clusters of related documents (small rectangles in different colors). Documents mentioning *Bukhari* are selected (surrounded by a yellow circle).

in which analysts are not tasked with examining specific people, places, and organizations. Instead, the analyst must initially learn about the different topics and contents within the data and decide what to investigate first.

Jigsaw initially did not have capabilities for finding themes or concepts in a document collection, and it was challenging for analysts to get started with an investigation. When we began working on the Contest dataset, we split the documents among ourselves and read all of them to find initial leads. We noted the need for a more global view of all the reports, one that could show which documents have been examined and that would allow the documents to be partitioned into groups.

To better assist analysts in browsing and understanding text documents in a more structured manner, we coupled the interactive visualizations in Jigsaw with automated computational analysis capabilities such as analyses of document summarization, document similarity, and document clustering. We integrated three different types of document summaries. We summarize single documents with a one-sentence summary by extracting the most important sentence from the document. We show the one-sentence summary in the Document View and also provide it as a tooltip wherever a document is shown by its icon. To summarize a collection of documents, we use word clouds (if there is enough space available, for example, in the Document View) or keyword summaries (if there is not a lot of space available, for example, in the Document Cluster View). The document summaries help analysts to quickly decide whether to read (a set of) documents in detail. Document similarity can be based on either the document text or the entities identified in or associated with the documents. Similarity helps analyst to understand whether a particular document is an outlier in a collection or part of a bigger theme (if there are similar documents). Document clustering can also be based on either the document text or the entities of the documents. It provides an overview of the document collection and helps analysts to explore the documents more systematically. (Additional details of the computational analyses are described in Görg et al.[2]) We integrated the computational analyses across a number of views in the system, as described below.

The Document Cluster View visualizes document clustering results and indicates which documents already have been read. One of Jigsaw's key capabilities, cross-filtering across views, becomes now even more powerful since it can highlight connections across entity-centric visualization, such as the List View, and document-centric visualization, such as the new Document Cluster View. Figure 2 shows an example in the context of the Challenge 2010 dataset. The List View (Figure 2(a)) shows locations, persons, and organizations connected to *Maulana Haq Bukhari*. *Karachi* and *Lyari Town* are the most connected locations, *Akram Basra* is the most connected person, and *Lashkar-e-Jhangvi* is the most connected organization. The Document Cluster View (Figure 2(b)) shows different clusters of related documents (small rectangles in different colors). Documents mentioning *Bukhari* are selected (surrounded by a yellow circle). The view illustrates that *Bukhari* is strongly connected to documents in the "town, house, basra" and the "bank, funds, transfer" clusters.

Additionally, we implemented a view that is tailored toward the representation of text analysis results. The Document Grid View can present, analyze, and compare a variety of document metrics, such as document similarity or sentiment. The view organizes the

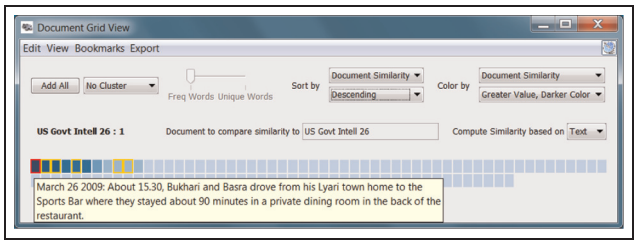


Figure 3. Document Grid View with the document (small rectangle) order and shading set to correspond to the documents' similarity to the selected report on *Bukhari*.

documents in a grid and provides an overview of all the documents' similarity to a selected document via the order and color of the documents in the grid representation. The Document Grid View in Figure 3 shows all documents of the Challenge 2010 dataset grouped and colored by similarity to a report on *Bukhari*. The tooltip displays the one-sentence summary of that report. The highlighted documents (yellow rectangle) also mention *Bukhari*. The computational analyses, in particular the document clustering, proved to be very useful in guiding the process of reading and making sense of the documents.

Do not neglect evidence marshaling

Taking notes of hypotheses and findings, tying them back to evidence, and keeping track of an investigation are important parts of analysis, especially for investigations that are carried out over a longer period of time. Jigsaw's support for these activities also evolved over time. When we participated in the VAST '07 Contest, the system did not provide any support for note-taking and we only relied on our manual notes on paper. After realizing this shortcoming, we developed the ShoeBox window to support evidence marshaling and note taking. We chose a structured approach for the view so that analysts could create hypotheses and then connect supporting as well as contradicting evidence to the hypotheses that were linked back to documents as provenance. The view provided a number of advanced features, such as "hypothesis-slides" that could be laid over one another similar to different graphic layers in image editing software such as Photoshop. This feature allowed analysts to compare hypotheses and ask "what if" questions, investigating different scenarios. However, the ShoeBox window was just too complex and did not allow analysts to take free-style notes in the way they would do on paper. Therefore, it was seldom used, and we abandoned it after some iterations.

As a replacement, we developed the Tablet as our new evidence marshaling tool. The Tablet adopts a minimalistic design, intending to offer greatest flexibility for visual thinking and sensemaking. Entities in Jigsaw's views can be directly added to the Tablet via popup menu commands. The added entities retain their original color coding according to their types. Analysts can also create their own items representing customized entities or events. Any two items or entities can be linked and the links can be labeled. Additional information about the items can be represented as post-it-notes (on a yellow background). Analysts can also create timelines and link entities or items to specific points on the timeline. All the visual items in the Tablet can be freely moved around and repositioned. With the Tablet, we took a free-style approach, mimicking analysts' note-taking behavior on paper. The basic constructs allow analysts to organize significant events into timelines, log and connect related people and organizations, and gradually build up hypotheses. The Tablet also allows the user to integrate bookmarks of views, as provenance or evidence, and the views can be re-instantiated at a later point to follow up on the original analysis. We used the Tablet in the VAST '10 Challenge submission and had much better results than we had with the ShoeBox. Figure 4 shows the initial ShoeBox (Figure 4(a)) and the new Tablet (Figure 4(b)).

Observations from our contest participation

Feedback from contest participation

Our participation in the VAST Contest/Challenges was extremely beneficial for us. It helped us to both improve and evolve the Jigsaw system, and it provided a hands-on learning experience through which we gained a better understanding of the analytical process in these types of investigations. This better understanding of the analytical process then directly guided some of our design decisions.

The availability of such large, high-fidelity datasets was invaluable in numerous ways, and it particularly allowed us to observe the utility of the different views in an actual investigation scenario. The feedback about our system we garnered from our own experience working on the contests was a useful complement to the feedback we received from other Jigsaw users. Participating in the contests allowed us to put ourselves into the shoes of an investigator and experience firsthand the virtues and shortcomings of our system. This knowledge motivated us to fix usability problems, create new operations and views, and consider future avenues for growth and expansion.

In particular, the live Contest session with professional analysts to which the winners of the VAST '07 Contest were invited was highly useful.[12] Working together with an analyst, seeing the analyst's perspective on an investigation, the kind of questions he asks,

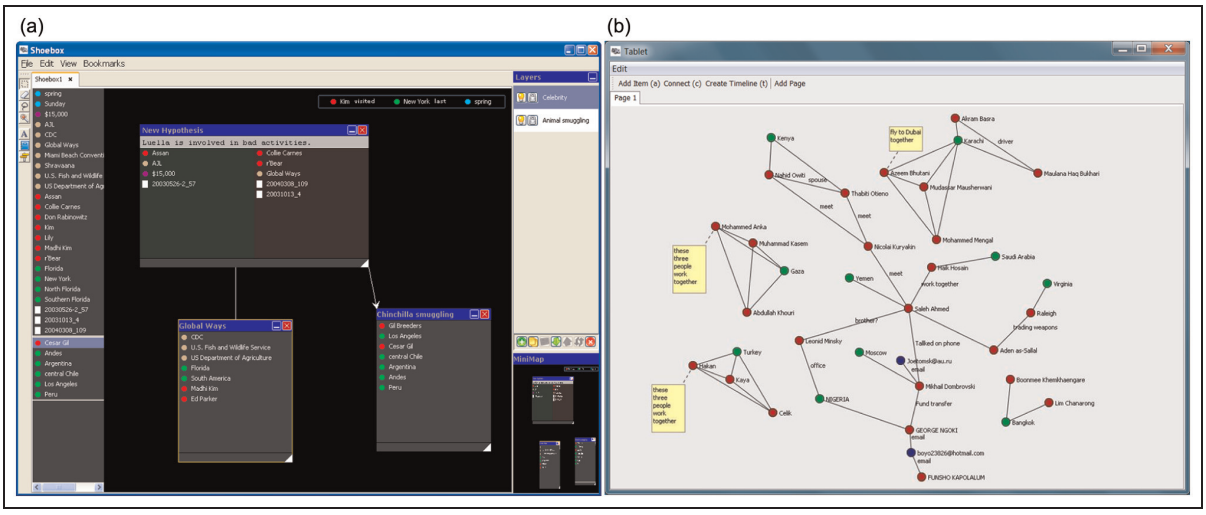


Figure 4. (a) The initial ShoeBox and (b) the new Tablet.

the kind of evidence he looks for, the way he builds trust in the data, and the way he interprets visual representations were very illuminating. It became clear which parts of the system aligned with the workflow of a professional analyst and which parts needed to be changed and improved.

The written feedback we received from our contest submissions was not as useful, however. Naturally, feedback that is based on a summary report and a video that only shows the "shiny" side of a system but not its shortcomings cannot have the same quality as feedback received from an interactive session with an analyst. Without observing the actual analytical process and the tool usage during the investigation, it is difficult for reviewers to provide accurate feedback that can drive the further development of a system.

Using contest datasets for evaluation

Our analysis activities in the VAST Contest/Challenges using the large examples with embedded ground truth exposed a number of shortcomings in the Jigsaw system, and thus, the activities functioned very much in a formative evaluation sense. The contest datasets have been used in class projects[13] and lend themselves for evaluation studies since answers can be easily graded because of the embedded ground truth. However, the contest datasets are too large and too complex to be used in lab experiments for formal evaluations of visual analytics systems. Such studies must be completed in a few hours, and the scope of the contest datasets does not fit this purpose. To evaluate the Jigsaw system in a formal experiment, we developed our own small dataset with embedded ground truth. The contest dataset and scoring scheme served as a model. Details of our system evaluation are described in Kang et al.[14] We do appreciate the effort it takes to prepare datasets with embedded ground truth, and we applaud the VAST Contest organizers' effort over the last 7 years.

Adoption and dissemination

Over the years, Jigsaw became better known as a visual analytics system through our participation in the Contest/Challenges. The publicity it received through our challenge participation and in particular the title of a winning entry of the VAST '07 Contest certainly helped to raise its visibility in the community. Analysts from a variety of areas contacted us and wanted to try the system on their own data.

We believe that two additional aspects of the system helped foster its adoption by other people: (1) an emphasis on usability and (2) simplified data import. We focused on two principles for achieving usability. First, Jigsaw uses only two primary interactive operations: a single click selects an entity or document and a double click expands the context of an entity or document (brings in additional entities or documents that are connected to it). Of course, there are a number of other advanced operations, but they are not required for a novice user to get started and they are not in the way of the simple interactions. Second, we consistently implemented brushing and linking across all the views to encourage the user to use the views together. The views are coordinated using an event mechanism: interactions with one view (selecting, adding, removing, or expanding entities) are transformed into events that are then broadcast to all other views. Thus, the views of the system stay consistent and provide different perspectives on the same data.

User support is crucial for the adoption of any system, and we employed a two-pronged approach. To

introduce novel users to the system, we created a number of training videos that guide the user step-by-step through the views and the analytical process. We also provide a number of example datasets that could readily be used. Additionally, we offer to help users to import their own data efficiently. The current version of Jigsaw does support the import of a number of file formats, including text, PDF, Word, Excel, and HTML. However, many different data formats exist, and we have found data import to still be a major hurdle.

Through its high visibility, Jigsaw gained popularity within the visual analytics research community. Researchers from various organizations used Jigsaw in conjunction with other tools to work on VAST Challenges. In the 2011 VAST Mini Challenge 3, for example, teams from the University of Konstanz and the City University London used Jigsaw in their visual analytical processes. The Konstanz team first identified candidate documents that were potentially relevant to the plot using keyword-based filtering and supervised machine learning classification. The researchers then used Jigsaw to explore the entities and documents within these candidate selections. Three views were particularly useful: the List View, the Document Cluster View, and the Document View. Similarly, the City University London team first used their own tool to perform keyword-based document filtering and to discover interesting organizations. With this initial set of leads, they used Jigsaw's Graph View to explore connected entities and the Document View to read documents. The Tablet View was useful for documenting how various entities were discovered in the analytical process.

These exemplary usage cases are consistent with our experience of using Jigsaw. Without a predefined theme or a set of keywords, it is usually difficult to identify a subset of documents related to the ultimate solution narrative. To our knowledge, few effective natural language processing tools can accomplish this task competently. The Document Cluster View in Jigsaw, backed by clustering algorithms, was moderately helpful on the challenge datasets. Jigsaw's strengths lie in its highly coordinated views, and both teams that used the system took advantage of this feature.

Outside of the VAST Contests, researchers have and are using Jigsaw for different purposes including targeting analysis and hypothesis generation based on police/intelligence case reports, comparing aviation documents,[15] understanding source code files for software analysis and engineering,[16] and genomics research based on PubMed articles.[17] Other domains or types of documents that have been analyzed with Jigsaw include investigative reporting, fraud, consumer reviews, academic publications, business intelligence, webpages, and blogs. Another article[18] reviews six individuals, including those from law enforcement and intelligence, who have used Jigsaw for periods from 2 to 14 months.

Synthetic datasets

The VAST Contest/Challenges are presented in a significantly different way than other visualization contests, such as the former InfoVis Contest: instead of using real-world datasets with open-ended questions, the organizers of the VAST Contest decided to create synthetic datasets with an embedded ground truth. Using synthetic datasets has a number of advantages: it is easier to ensure that a dataset has the right scope since the organizers can decide how large and complex the dataset should be, and the ground truth allows the organizers to better judge the submitted entries on their accuracy. Additionally, synthetic datasets often motivate the participants since they know that a solution does exist, and they also know that the analysis is feasible for students and researchers, not requiring the knowledge and background of a professional analyst. However, we also noted a few shortcomings of the text-based synthetic datasets used in the VAST Contests. These datasets are a mix of "real" text documents and "contrived" text documents. Real events and people are not likely to be involved in the embedded ground truth, and therefore, we excluded (sometimes consciously, sometimes maybe subconsciously) documents that mentioned real events or real people. The synthetic datasets are very useful for promoting and demonstrating a visual analytics system. Having a dataset that is clearly not a toy example but also not too large, and knowing an interesting story within it, lends itself for good system demonstrations.

Scalability

Scalability is an important aspect of any visual analytics system. Jigsaw evolved over the years from using an in-memory data model, capable of handling a few thousand documents and tens of thousands of entities, to an architecture using a database framework, capable of handling tens of thousands of documents and millions of entities. This evolution was driven by the demand of real-world clients and their applications and not by our participation in the VAST Contests. The synthetic datasets used in the contests were not large enough to motivate the move from an in-memory model to a database model. This decision of the VAST Contest organizers is understandable given the multiple purposes that the contest data were serving. Keeping the datasets at more modest sizes also encouraged more teams to work on the problems and submit entries.

Conclusion

A primary task in the VAST Contest/Challenges is to "connect the dots" or "put the pieces together." This task aligns very closely with the primary purpose of Jigsaw, and we found the existence of the contest datasets very beneficial for the development of the system. In this article, we have described how Jigsaw evolved from a very visualization-centric system to a balanced visual analytics system that provides and integrates both computational text analyses and interactive visualizations of entities and documents. Our experiences from the VAST Contest and Challenges influenced our design decisions and guided us throughout this process. Additionally, we discussed our observations from participating in the Contest and Challenges, including feedback from our participation, lessons on evaluation, adaption, dissemination, and properties of synthetic datasets.

Acknowledgements

Many other students have contributed to Jigsaw's development and our participation in the VAST Challenges, including Meekal Bajaj, Elizabeth Braunstein, Jaegul Choo, Alex Humesky, Jaeyeon Kihm, Vasilios Pantazopoulos, Neel Parekh, Kanupriya Singhal, Gennadiy Stepanov, and Sarah Williams. We also would like to thank the organizers of the VAST Contests and Challenges for their continued effort in creating the datasets and judging the submissions.

Funding

This research is based upon the work supported in part by the National Science Foundation via Awards IIS-0414667, IIS-0915788, and CCF-0808863; by the National Visualization and Analytics Center (NVAC), a US Department of Homeland Security Program; and by the US Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0001.

References

1. Stasko J, Görg C and Liu Z. Jigsaw: supporting investigative analysis through interactive visualization. *Inform Visual* 2008; 7(2): 118–132.
2. Görg C, Liu Z, Kihm J, et al. Combining computational analyses and interactive visualization for document exploration and sensemaking in Jigsaw. *IEEE T Vis Comput Gr*, in press.
3. Thomas JJ and Cook KA. *Illuminating the path*. Washington, DC, USA: IEEE Computer Society, 2005.
4. Görg C, Liu Z, Parekh N, et al. Jigsaw meets Blue Iguanodon—the VAST 2007 contest. In: *IEEE VAST*, Sacramento, CA, October 2007, pp. 235–236. Washington, DC, USA: IEEE Computer Society.
5. Görg C, Liu Z, Parekh N, et al. Visual analytics with Jigsaw. In: *IEEE VAST*, Sacramento, CA, October 2007, pp. 201–202. Washington, DC, USA: IEEE Computer Society.
6. Liu Z, Görg C, Kihm J, et al. Data ingestion and evidence marshalling in Jigsaw. In: *IEEE VAST*, Salt Lake City, UT, October 2010, pp. 271–272. Washington, DC, USA:IEEE Computer Society.
7. Braunstein E, Görg C, Liu Z, et al. Jigsaw to save Vastopolis—VAST 2011 Mini Challenge 3 Award: "Good Use of the Analytic Process." In: *IEEE VAST*, Providence, RI, October 2011, pp. 323–324. Washington, DC, USA: IEEE Computer Society.
8. Stasko J, Görg C, Liu Z, et al. Jigsaw: supporting investigative analysis through interactive visualization. In: *IEEE symposium on visual analytics science and technology 2007 (VAST 2007)*, Sacramento, CA, October 2007, pp. 131–138. Washington, DC, USA: IEEE Computer Society.
9. Wattenberg M and Viégas FB. The word tree, an interactive visual concordance. *IEEE T Vis Comput Gr* 2008; 14(6): 1221–1228.
10. Cunningham H, Maynard D, Bontcheva K, et al. *Text processing with GATE* (Version 6). Gateway Press CA, 2011.
11. Ratinov L and Roth D. Design challenges and misconceptions in named entity recognition. In: *CoNLL*, Boulder, CO, June 2009, pp. 147–155. Stroudsburg, PA, USA: Association for Computational Linguistics
12. Plaisant C, Grinstein G, Scholtz J, et al. Evaluating Visual Analytics: The 2007 Visual Analytics Science and Technology Symposium Contest. *IEEE Computer Graphics & Applications* 2008; 28(2): 12–21.
13. Whiting MA, North C, Endert A, et al. VAST contest dataset use in education. In: *IEEE symposium on visual analytics science and technology 2009 (VAST 2009)*, Atlantic City, NJ, October 2009, pp. 115–122. Washington, DC, USA: IEEE Computer Society.
14. Kang Y-A, Görg C and Stasko J. How can visual analytics assist investigative analysis? Design implications from an evaluation. *IEEE T Vis Comput Gr* 2011; 17(5): 570–583.
15. Pinon OJ, Mavris DN and Garcia E. Harmonizing European and American aviation modernization efforts through visual analytics. *J Aircraft* 2011; 48: 1482–1494.
16. Ruan H, Anslow C, Marshall S, et al. Exploring the inventor's paradox: applying Jigsaw to software visualization. In: *ACM SOFTVIS*, Salt Lake City, UT, October 2010, pp. 83–92. New York, NY, USA: ACM.
17. Görg C, Tipney H, Verspoor K, et al. Visualization and language processing for supporting analysis across the biomedical literature. In: *Knowledge-based and intelligent information and engineering systems* Setchi R, Jordanov I, Howlett RJ and Jain LC (eds). LNCS Berlin Heidelberg: Springer, 2010.
18. Kang Y-A and Stasko J. Examining the use of a visual analytics system for sensemaking tasks: case studies with domain experts. *IEEE T Vis Comput Gr* 2012; 18(12): 2869–2878.

Value-Driven Evaluation of Visualizations

John Stasko
School of Interactive Computing
85 5th St., NW
Georgia Institute of Technology
1-404-894-5617
stasko@cc.gatech.edu

ABSTRACT
Existing evaluations of data visualizations often employ a series of low-level, detailed questions to be answered or benchmark tasks to be performed. While that methodology can be helpful to determine a visualization's usability, such evaluations overlook the key benefits that visualization uniquely provides over other data analysis methods. I propose a *value-driven evaluation* of visualizations in which a person illustrates a system's value through four important capabilities: minimizing the time to answer diverse questions, spurring the generation of insights and insightful questions, conveying the essence of the data, and generating confidence and knowledge about the data's domain and context. Additionally, I explain how interaction is instrumental in creating much of the value that can be found in visualizations.

Categories and Subject Descriptors
H.5.2 [**Information Interfaces and Presentation, (e.g., HCI)**]: User interfaces – *Evaluation/Methodology*.

General Terms
Measurement, Design, Human Factors, Theory.

Keywords
Data visualization, value, evaluation, interaction.

1. INTRODUCTION
The topic of evaluation (broadly considered) has risen in prominence in the data visualization research community over the past few years. To some degree, the novelty of creating new techniques and systems has worn off, or at least researchers have broadened their views beyond just creating new visualization techniques. The community has become more reflective and introspective, and thus evaluation seems to be a topic on everyone's minds currently.

What does "evaluation" mean in the context of visualization, however? What are we evaluating? For what purposes are we evaluating? I think these questions are more subtle than one would immediately surmise, and the answers are nuanced.

BELIV'14, November 10, 2014, Paris, France.

ACM 978-1-4503-3209-5/14/11 …$15.00.
http://dx.doi.org/10.1145/2669557.2669579

One potential angle of evaluation is to improve the techniques and systems one builds. That is, a developer of a new system should evaluate it and find the embedded problems and faults in order to help improve the system and make it better. This activity can be iterated repeatedly and is a fundamental component of formative evaluation that one encounters in the area of human-computer interaction.

A second style of evaluation is to compare two specific approaches to each other. Is technique A or technique B a better approach for a given problem? The specificity of this type of evaluation is appealing in many ways, but it is frequently quite difficult to conduct such an evaluation in the field of data visualization because very few techniques or systems are built for the exact same purpose, domain, and the same type of data. Examples of this type of evaluation do exist [14,29] but it is typically very difficult to "compare apples to apples."

Another angle of evaluation is more general than these first two. When a person develops a new visualization technique or system, there is a fundamental desire to determine whether it is any *good*. Simply put, is the technique useful and beneficial? Researchers want to show the value and utility of their new ideas, and they seek methods to answer those questions.

This notion of showing or identifying utility and value is one way to slightly recast the challenge of evaluation in data visualization. Rather than thinking about evaluating techniques and systems, one could focus on identifying the potential *value* of a system. Hence, the evaluation of a technique or system becomes a process where one identifies and illustrates its value.

I also believe that the notion of identifying value has a broader scope than evaluation does. Researchers in other fields probably will not be so interested in the evaluation of a particular visualization system. Conversely, showing the value of a visualization technique can be extremely important at a more general scale. Many approaches to analyzing and presenting data exist; visualization is just one of many approaches available to people and organizations today. Identifying the value of visualization better is vital to educating people not so familiar with the domain and convincing them of the potential impact that a visualization approach can achieve.

In the remainder of this article, I explore why the process of "evaluation" is more problematic in visualization research than in human-computer interaction research, on the whole. I recast the evaluation of a visualization as identifying its value, and I develop a qualitative formula for determining a visualization's value. Although the equation is more descriptive than prescriptive, it should help researchers to more accurately and objectively assess and identify the value of their new ideas, and it should help to more specifically communicate visualization's utility to those people less familiar with the area. Finally, I explain the

fundamental importance of interaction in the value equation even though it is still the less understood half of data visualization.

2. BACKGROUND

After ten years of research focused on technique and system development, the field of information visualization began to consider issues involving evaluation more deeply around 2000. That year, a special issue of the journal *International Journal of Human-Computer Studies* [7] was devoted to empirical studies of information visualizations. Subsequently, the initiation of the BELIV workshop in 2004, whose focus is solely on evaluation in visualization, further stimulated research into this topic. Plaisant's keynote address there and accompanying article [21] outlined a number of challenges in evaluating information visualization including the need to match tools with users, tasks, and real problems, as well as difficulties in improving user testing. She called for repositories of data and tasks and a broader notion of evaluation including case studies and success stories. Scholtz [26] later argued for a metrics-driven evaluation methodology in visual analytics that moves beyond usability and includes evaluations of situation awareness, collaboration, interaction, creativity, and utility.

A number of different methodological approaches have been proposed for visualization evaluation. The insight-based evaluation methodology [25] attempts to count and document the insights gained during an analysis session. An insight is defined as an individual observation about the data by a participant, a unit of discovery. The MILC evaluation methodology [28] advocates performing long-term, multidimensional case studies of deployed systems in real use outside the laboratory. This approach follows a more ethnographic style and centers on observational evidence about the utility of a system.

Following a 2007 Dagstuhl seminar on information visualization, Carpendale authored a book chapter [5] focusing on many issues involving visualization evaluation. She described different methodological approaches that could be taken, along with the benefits and challenges inherent in each. Based on a meta-analysis of over 800 research articles, Lam et al. [15] describe seven canonical scenarios of empirical studies in information visualization. For each scenario, the authors describe goals and outputs, evaluation questions, and they provide methods and examples. A key notion of their work, also emphasized in this article, is understanding the purpose of an evaluation. Subsequently, Isenberg et al. [12] performed a similar study for scientific visualization articles.

Moving beyond evaluation, other researchers have carried out research explicating the value of visualization to cognition. Larkin and Simon [16] describe how visualization helps people think by acting as a temporary storage area (extra memory), by transforming cognitive challenges into perceptual actions, and by engaging people's strong pattern matching abilities, among other benefits. Norman [18] discusses the value of visualization in helping people to think and explains how external representations aid information access and computation. He specifically describes the importance of matching representations to the task at hand in order for visuals to provide the best cognitive value. Card, Mackinlay, and Shneiderman's book of collected information visualization readings [4] contains a first chapter with an extensive discussion on the benefits of visualization. The editors present a model of knowledge crystallization and list ways that visualization helps amplify cognition, such as by reducing searches for information, enhancing the recognition of patterns, and enabling perceptual inference operations.

With respect to analyses of visualization focusing on its value to people, van Wijk [33] presents a more formal, theoretical model of value. He first develops an equation for the central process in visualization, which describes time-varying images that are dependent upon the data and the visual specification. Knowledge is then defined by a person's initial knowledge plus the knowledge gained at each step as the person perceives the visualization image. Knowledge can be affected by interactive exploration as well. Next, he describes the costs of visualization which include initial development costs, initial costs per user and session, and perception and exploration costs. Finally, his economic model defines that the total (cognitive) profit is equal to the value minus those costs. Fekete et al.'s book chapter on the value of visualization [9] articulates multiple ways that visualization assists data understanding. The authors discuss cognitive and perceptual support, further describe van Wijk's economical model of value, and provide multiple "success story" examples that help illustrate visualization's utility.

3. VALUE-DRIVEN EVALUATION

In the field of human-computer interaction, evaluation often equates to assessing the usability of a system or interface. Through sessions with a series of benchmark tasks to be performed, evaluators determine whether potential users can employ a system to achieve desired results. That is, an evaluator seeks to learn whether people can effectively and efficiently use a system to successfully perform tasks. Such evaluation is a key aspect of user-centered design and it has become pervasive throughout software development, particularly for interactive systems.

Unfortunately, this type of benchmark task-focused evaluation is simply not sufficient for evaluating visualization systems, even though it is often what one observes when reading research papers within the discipline. This type of evaluation can help determine whether a visualization is learnable and comprehensible, and whether a person can use it to answer specific questions about a data set. However, this approach fails to examine some of the larger benefits of visualization that go beyond just answering specific data-centered questions. Visualization should ideally provide broader, more holistic benefits to a person about a data set, giving a "bigger picture" understanding of the data and spurring insights beyond specific data case values.

To help understand this premise, consider a data set of information about different car models and each car's attributes such as price, miles per gallon, horsepower, and so on. One could build a visualization of this data set and "evaluate" the visualization by conducting a user study where participants answer specific questions (i.e., perform benchmark tasks) about the data such as

- Which cars have the best miles-per-gallon?
- How much does a Ford Taurus cost?
- Is there a correlation between car weight and torque?
- How many different countries manufacture cars?
- What is the range of car's horsepower?

While it is important for the visualization system to support answering such questions thus illustrating that it is

comprehensible and usable, a person could relatively easily gain those answers via a spreadsheet, query language, or statistics package as well.[1] Support for answering those types of questions is not a unique benefit of visualization. An evaluation that focuses only on these types of tasks simply fails to adequately assess the primary and unique benefits of visualization that go beyond simple data-driven queries.

What are the broader benefits that visualization provides? This question is at the heart of the approach I advocate for evaluating visualization systems. It involves a fundamental examination of the benefits that are more unique and specific to visualization, compared to other types of data analysis. The approach that I advocate focuses on identifying the *value* that a visualization provides. This value goes beyond the ability to support answering questions about data—it centers upon a visualization's ability to convey a true *understanding* of the data, a more holistic broad and deep innate sense of the context and importance of the data in "the big picture."

To be more specific, I have developed a simple equation that characterizes the value (**V**) of a visualization:

$$\mathbf{V = T + I + E + C}$$

Below, I elaborate on each of the four components of the value equation and then provide two examples to show how it applies to existing visualizations.

The **T** in the value equation represents:

> A visualization's ability to minimize the total **time** needed to answer a wide variety of questions about the data.

Effective visualizations allow a person to identify many values from a data set and thus answer different questions about the data simply by viewing the visualization or, at least, by interacting with the visualization and inspecting the resulting views. Rather than have to learn query languages and tools and issue syntactic requests, a person merely interacts with an interface using direct manipulation to select visual items or update the view to show the desired information. Effective visualizations also excel at presenting a set of heterogeneous data attributes in parallel. They make accessing both quantitative and nominal data take the same relatively low amount of effort.

The types of "low-level" questions about data that visualization can assist with have been described in earlier research [1,31,35]. They include tasks such as retrieving values, finding extrema, characterizing distributions, and identifying correlations, among others.

The **I** in the value equation represents:

> A visualization's ability to spur and discover **insights** and/or **insightful questions** about the data.

Effective visualizations allow a person to learn about and make inferences from a data set that would be much more difficult to achieve without the visualizations. This notion of knowledge gained goes beyond simply reading relevant data values. It often involves knowledge and inferences that are acquired by viewing combinations, alignments, aggregations, juxtapositions, and particular unique representations of the data.

The notion of insight has been a key component of information visualization for many years. The insight-based evaluation methodology [25] by Saraiya, North, and Duca, discussed earlier, provided an initial formal notion of insight. North [19] defined an insight as an individual observation about the data by the participant, a unit of discovery, and characterized each as being complex, deep, qualitative, relevant, and unexpected. This characterization, in particular the perception of insights being unexpected, seems to align with a notion of spontaneous insights, the "Aha!" moments when people finally see solutions to problems.

I do not require the notion of unexpectedness for the **I** component of the value equation. Instead, my view of insight aligns with that described by Chang et al. [6], and refers to the acts of knowledge-building and model-confirmation. Insight is like a substance that people acquire with the aid of a system. It is a by-product of exploration, not a task-driven result of analysis [38].

The **E** in the value equation represents:

> A visualization's ability to convey an overall **essence** or take-away sense of the data.

Effective visualizations allow a person to gain a broad, total sense of a potentially large data set, beyond what can be gained from learning about each individual data case and its attributes. In essence, to borrow from a well-known phrase, "the whole (understanding) should be greater than the sum of the parts". An effective visualization should convey information about the totality of data being presented, effectively the "big picture."

The notion of providing an overall sense of a data set is often implemented through visualization design principles such as "overview and detail" or "focus + context." Shneiderman's oft-repeated information visualization mantra [27], "Overview first, zoom and filter, details on demand", begins with an overview. The **E** component of the value equation goes beyond supplying an overview of the data set, however. A visualization with high utility should convey an overall essence of the data, its unique and most important characteristics, and the primary knowledge to be gained from the data.

The **C** in the value equation represents:

> A visualization's ability to generate **confidence**, knowledge, and trust about the data, its domain and context.

Effective visualizations allow a person to learn and understand more than just the raw information contained within the data. They help promote a broader understanding of the importance of the data within its domain context. Furthermore, effective visualization promotes a viewer's sense of confidence, and thus trust, of the data.

The knowledge and sense of confidence or trust can apply both to the visualization creators and the visualization consumers. Visualizations, and more specifically the visualization construction process, are a wonderful way to identify embedded problems in a data set such as missing, erroneous, or incomplete values. Similarly, viewing a data set with an effective visualization can highlight areas where more data may be needed

[1] Of course, each of these tools has a learning curve, but once understood they can be effective in this role.

or can signify the importance of adding and exploring other data sets.

This equation for identifying the value of a visualization is still just a qualitative metric. I have not developed quantitative measures of each of the four key components, nor the factors to determine how much each component contributes to the overall value. As such, the equation serves more as a *descriptive* aid than a *prescriptive* one.

One could imagine a variety of mechanisms for more precisely determining the benefits of a visualization toward each of these four components. The most basic method would be providing a thorough and clear explanation of a visualization, possibly including demonstrations of use through videos or direct hands-on interaction. A second method could be offering descriptions of detailed scenarios of use that illustrate how to apply the system in context and how it would be beneficial to potential relevant users. Going even further, the developers of the visualization could deploy it in the field and then report on its trial usage for an extended period of time. Furthermore, experiments and user studies that go beyond employing simple sets of benchmark tasks could be useful for helping to determine value.

In the remainder of this section, I examine two example visualization applications and illustrate how each provides value via the introduced equation. Here, I simply describe how each provides the four constituent components, but future work can take on the task of making these assessments even more descriptive and precise.

3.1 Map of the Market

The first example, shown in Figure 1, is the Map of the Market tool (http://www.marketwatch.com/tools/stockresearch/marketmap) that uses the treemap visualization technique to portray companies' performance on the stock market. It includes interface controls to search for a particular company and to adjust the period being shown, for example, showing year-to-date versus daily performance. The original version of this visualization was created by Wattenberg [34] for the smartmoney.com website. It used more saturated colors than the current version and provided more interactive controls for showing top gainers and losers.

T: The Map of the Market supports rapidly answering a variety of questions about stocks' performance simply by mousing over an item, clicking on it, or searching for a company. The viewer can learn how particular companies have performed, how bigger or smaller companies have performed, which companies have gained or lost significant amounts, how different sectors have performed, and so on. Answering these questions is quick and easy because it requires observation and simple interface actions. Note, however, that not every type of query can be answered easily. For instance, it may be difficult to determine which company has gained or lost the most because multiple companies will exhibit similar shades of a color.

I: The visualization supports the development of insights and insightful questions as the viewer notices patterns, trends, or outliers. For instance, one sector may have performed well during the period but another poorly, or perhaps large companies performed well while small did not. The viewer may wonder why such a result has occurred. By comparing the views of year-to-date and the prior 52 weeks performance, a person may notice different companies whose stock is trending upward or

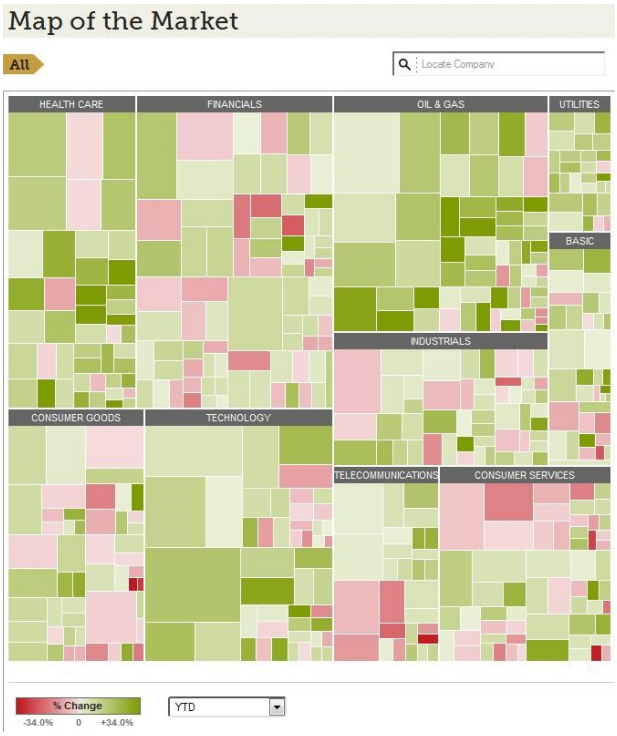


Figure 1. The Map of the Market showing stock performance (http://www.marketwatch.com/tools/stockresearch/marketmap)

downward.

E: Beyond the individual company performances, viewing the complete treemap gives a person a broad perspective on how the market has done as a whole. By observing the ratio of green to red cells, one can estimate the general performance. Similarly, on days when the market has a tremendous gain or crash, it communicates an overwhelming view of one bright color. Outliers also can emerge from these types of views. For example, during the 777 point drop on September 29, 2008, the former smartmoney.com site showed an overwhelming view of bright red except for one small green cell in the Basic Materials sector, a gold stock.

C: Viewing the market visualization from different periods gives a person a sense of the level of up and down swings that stocks typically make. Observing the sectors with the largest rectangles communicates where the largest companies reside, and thus informs the viewer about American companies on the whole. Effectively, this visualization goes beyond simply communicating stock performance data. It also begins to inform the viewer about companies, markets, and the economy.

3.2 CiteVis

The second example, shown in Figure 2, is the CiteVis application (http://www.cc.gatech.edu/gvu/ii/citevis/) that presents citation data about papers appearing at the IEEE Information Visualization Conference. The application was created by my students and me to help researchers learn about and understand the histories of citations to and among these articles [30]. It represents the papers for each year as a row of circles and uses

interaction to show specific paper-to-paper citations rather than drawing edges between them. The application also allows a viewer to search for papers from specific authors or about specific topics.

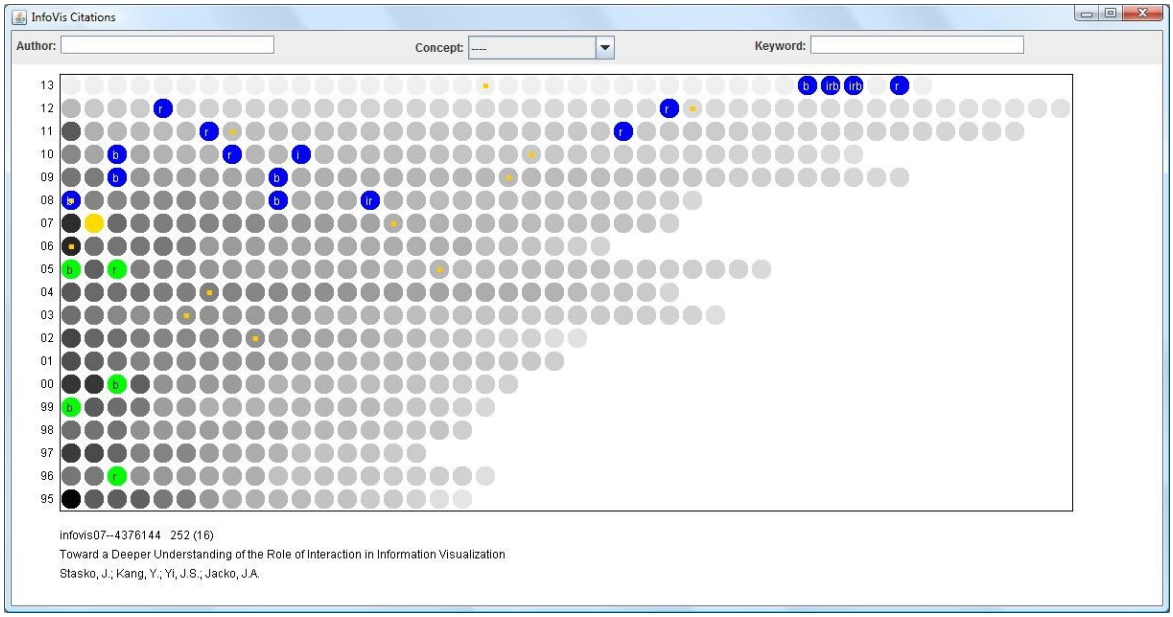


Figure 2. The CiteVis application showing InfoVis paper citations (http://www.cc.gatech.edu/gvu/ii/citevis/)

T: Just by viewing the CiteVis application, a person can quickly answer questions and learn about the number of papers each year and how the number of citations change going back in time. Furthermore, through simple interactions, many more questions can be answered and information acquired. By mousing over a paper's circle, the viewer learns its title, authors, number of total and internal (within the conference) citations that the paper has gained, and the internal papers it cites and is cited by. Through the menus and search fields at the top, viewers can observe the papers of a particular author or on a particular topic. Mousing over the darkest circles tells the viewer which papers have been cited the most. Observing the small yellow dots informs the viewer how the Best Paper Award winners have performed in terms of citations. Again, it is important to stress that no query language need be learned to acquire all this different information. Direct observation coupled with simple interface interactions answer the viewer's questions and communicate a wide variety of aspects of this data set very rapidly and easily.

I: The CiteVis application helps a person learn insights about the data. For example, just by examining the default view, one can notice the four highly cited (dark circles) papers from the year 2000 or the large set of more highly cited papers in 2004. Clearly, these were two good years for impactful, high quality articles. A viewer can notice how the Best Paper winners are not always among the most highly cited papers from a year. Issuing queries to spotlight papers mentioning "evaluation" or "user study" in their abstract highlights how the early years of the conference had relatively little work on that topic, but it has grown strongly since. When we explored the visualization one day, we noticed how many of the least cited papers to the right side of the view seemed to be the application/design study style of papers. We were curious if this is actually the case, but because the data set does not include the paper type, that answer cannot be conclusively determined. However, it shows how the visualization helped develop an insightful question. We speculate that many of these observations and insights would have been difficult to recognize simply by accessing the data through a query interface or tables and spreadsheets containing all the data values.

E: The CiteVis visualization conveys a global sense of how the conference has grown and how citation patterns change over time. There are fairly recent papers that have high citation counts, so it is not only older papers with that characteristic. The visualization conveys the essence of the data set, its overall trends, patterns, and peculiarities.

C: To create the CiteVis application, we gathered data from online digital libraries and we manually examined the papers from each year and logged relevant information. Upon attempting to run the visualization for the first time, we quickly learned of missing papers and faulty data values because the system would crash upon certain interactions. Similarly, we observed papers that supposedly cited a paper from the future, which clearly cannot happen and again pointed out errors in the data. Beyond inspiring more confidence in the data by highlighting errors, the visualization also informs the viewer about the field of information visualization itself, what subtopics are important, which authors have written high-impact papers, and how the field is growing. The visualization communicates knowledge about the domain and its context beyond simple citation counts on papers.

4. THE IMPORTANCE OF INTERACTION

The previous section explains how value is manifest in visualization through the identified four components. In this section, I dig deeper to uncover how that value is frequently generated. To begin, let us consider different design paradigms commonly employed in visualization today. Suppose that an organization has a data set with multiple variables (attributes) per data case. To create a visualization to help explore, analyze, and communicate that data, three fundamental visualization design paradigms could be used.

1. Create a rich visualization employing many graphical objects and a diverse set of visual properties that completely represents the entire data set. Effectively, this design packs all the data into one, usually complex, view or representation.
2. Provide multiple, coordinated views of the data set, where each view provides a different perspective on the data, perhaps just focusing on certain attributes of the data.
3. Represent only some of the data cases and/or attributes of the data set initially, but employ interaction to allow the viewer to change the cases, attributes, and representation being shown.

The first scenario is very common today and is exemplified by infographics so pervasive throughout print media and the web. These designs are static and often physically large, in order to include all aspects of the data set being shown.

As data sets become larger with more variables per case, it becomes increasingly difficult to rely on the first design scenario, however. There simply may be too much to show, or the representation becomes so complex that it is difficult for people to interpret. Thus, visualization designers often move to using the second or third design paradigms, or perhaps a hybrid of the two.

Both the Map of the Market and CiteVis applications discussed in the previous section are examples of the third paradigm. The key differentiator in this design is the use of interaction in the visualization. Interaction differentiates these types of visualizations from static infographics. Interaction, when applied well, allows a person using a visualization to engage in a rich dialog with the data, to effectively conduct a fruitful conversation including a variety of questions and answers. As a person considers aspects of the data and forms new schemas and models of the data's meaning, interaction allows the person to change perspective and address new inquiries that arise.

Representation and *interaction* are often seen as the two key components of visualization, but I believe it is fair to say that representation has received the majority of attention throughout the visualization research community. Interaction, while important, has not received as much focus. As discussed above, many infographics, which some people (erroneously) equate to data visualization, do not even include interactive capabilities.

What, therefore, is the role of interaction in visualization? Is it an equal peer with representation, playing just as important a role in providing value for analysis and presentation? Alternately, is interaction subordinate to representation in the visualization equation, serving merely as a facilitating mechanism to move between different representations? This is an interesting philosophical question. I tend to view interaction as a significant and vital component in the equation, playing an equally important role in providing effective, valuable understanding of data to people.

To explore interaction's role in the value equation more closely, it is helpful to examine the many different aspects of interaction. When a person interacts or seeks to interact with a visualization, s/he does so for a purpose and to gain further understanding. Thus, one important component of interaction is the user's *intent*, the reason why they interact. Yi et al. [37] studied over a hundred visualizations and examined how interaction functioned in each. They developed a list of seven types of intent that covered the vast majority of uses of interaction. They found that systems provide and people interact with visualizations to select, explore, reconfigure, encode, abstract/elaborate, filter, and connect.

Each of these intents facilitates knowledge acquisition from a visualization and thus plays a key role in the value equation. As shown by the two example systems in section 3, interaction is often crucial to simplified, fast answers to varieties of questions (**T**) that can be posed to a visualization. Interaction likewise enables the generation of insights and insightful questions (**I**) because new views of the data provide new mental perspectives for a person. Interaction may play a less important role in conveying the essence of a data set (**E**), but as data set sizes increase, a visualization simply may not be able to provide an adequate summary or overview, and hence may require interaction to communicate the "big picture." Finally, interaction facilitates confidence about the data, its context, and domain (**C**) by allowing a person to explore all dimensions and aspects of the data.

While intent is one aspect of interaction, how that intent is carried out operationally is yet another. That is, the set of interactive operations provided by a system also contributes fundamentally to its value. Prototypical lists of such operations are discussed in many articles and books [10,13,27] and include selecting items for identification and manipulation, navigation such as zooming, filtering items or values, and coordinating views, among others. Curious whether these operations are actually manifest in real systems, my doctoral student Charles Stolper recently conducted a survey of hundreds of visualization applications and systems found online. He logged the types of interactive operations each provides and found that the following small set of operations are pervasive and comprise the majority of interactions provided by the systems:

- Selection to access data details, including mouse-over "tooltip" or "balloon"-style details
- Navigation, including panning and zooming
- Brushing and linking across views or representations to show connections and relationships

We were surprised that these three interaction operations so dominate interactive visualization applications. We expected more diversity in systems because research has demonstrated many new types of interactive operations. Thus, we wondered if this lack of diversity means that either a) practitioners only feel that this small set of operations are truly useful or b) the visualization libraries and toolkits used to build the systems implicitly limit interaction to these types of operations.

Nonetheless, this observation shows that there is room to innovate on interaction within visualization. For example, the use of lenses or toolglasses [3] to provide different views of data is an old idea that is rarely seen in visualization applications. Other systems, such as Dust and Magnet [36] and Kinetica [22], represent data cases as physical objects that can be dragged and pushed to expose their underlying attributes. Furthermore, interaction can be used to implement semantic operations within a domain, such as the OnSet system's use of drag-and-drop style interaction to perform union and intersection operations on set-typed data [23].

Yet another aspect of interaction concerns the interface mechanisms facilitated by existing hardware used to implement

interaction operations. The vast majority of visualization applications run on desktop computers with a keyboard, mouse, and monitor(s). They provide WIMP (window, icon, menu, pointer)-style interfaces with many controls and (often small) graphical objects representing the data. Hence, the use of a mouse is crucial and advantageous to manipulate these detailed controls and objects. Today, as more people transition to using tablets where finger or pen-touch are the only modes of interaction, visualization interfaces must transition to these modes as well [17]. A few research projects have explored visualization interaction via pen and finger touch [2,11,24], and they illustrate how new ideas are necessary to continue to provide value under this different hardware platform.

The goal of this section was to illustrate how vital interaction is to the value provided by visualization systems, but also how interaction is still the less understood half of the overall contribution. The research community has recently been making strides toward understanding interaction better. For instance, Elmqvist et al. [8] discuss the importance of interaction in visualization, focusing primarily on how fluid interaction can engage a person in a type of cognitive flow that enhances analysis and productivity. Furthermore, we need to move toward a science of interaction [20] including theories and testable premises that explain more fully how interaction contributes value to people's use of visualization systems. Fundamentally, interaction remains a relatively under-utilized component of visualization for providing greater analytical value to people.

5. CONCLUSION

In this article, I propose a new style of value-based evaluation of visualization techniques and systems. This approach hinges upon evaluators identifying the *value* that a visualization provides, beyond its ability to provide answers to basic queries about the data it represents. The technique defines a visualization's value through four constituent capabilities: T – minimizing the time to answer diverse questions about the data, I – spurring the generation of insights and insightful questions about the data, E – conveying the overall essence of the data, and C – generating confidence and knowledge about the data's domain and context.

This value-based evaluation methodology is presently still a qualitative, descriptive approach rather than a quantitative and more prescriptive technique. One might argue that for the approach and the value equation to become practical tools for evaluating visualizations' value, they need more of a quantitative characterization. While that is true at a pure sense, the equation as-is still provides a framework for visualization researchers to communicate the value of their creations, and it provides four dimensions that can serve as goals for researchers developing new evaluation methodologies. For instance, what type of evaluation could measure the ability of a visualization to convey the essence or take-away sense of a data set?

My approach in this article has largely been utilitarian, focusing on the analytic value of visualizations. Visualizations also can provide aesthetic impact and value [32]. Future work might explore bringing this notion to the value equation as well.

This article also advocates further study and use of interaction to provide even more value with visualization. Creative new methods of interaction from the research community have yet to be widely used in practice, and new platforms such as touch-based screens and tablets require new types of interactive operations.

Visualization goes beyond simply answering specific questions about a data set. It also helps to generate insights and insightful questions about the data that would be difficult to identify so easily using other methods of analysis. Visualization additionally conveys the "big picture" essence of a data set, and it facilitates generating confidence and learning about the domain of the data and its context. We need evaluation methodologies that focus on identifying the most beneficial, unique capabilities that visualization provides.

6. ACKNOWLEDGMENTS

This article was motivated by my EuroVis '14 Capstone address (http://vimeo.com/98986594). I thank my students for discussions helping in the development of these ideas. The research has been supported by the US Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0001, DARPA's XDATA program, and the National Science Foundation via awards CCF-0808863, IIS-0915788, and IIS-1320537.

7. REFERENCES

[1] Amar, R., Eagan, J., and Stasko, J. 2005. Low Level Components of Analytic Activity in Information Visualization, In *Proceedings of the IEEE Symposium on Information Visualization* (Minneapolis, MN, October 2005). InfoVis '05. 111-117.

[2] Baur, D., Lee, B. and Carpendale, S. 2012. TouchWave: Kinetic Multi-touch Manipulation for Hierarchical Stacked Graphs. In *Proc. of ACM Interactive Tabletops & Surfaces* (Cambridge, MA, November 2012). ITS '12. 255-264.

[3] Bier, E.A., Stone, M.C., Pier, K., Buxton, W., and DeRose, T.D. 1993. Toolglass and magic lenses: the see-through interface. In *Proceedings of the ACM Conference on Interactive Graphics* (Anaheim, CA, August 1993). SIGGRAPH '93. 73-80

[4] Card, S., Mackinlay, J., and Shneiderman, B. 1999. *Readings in information visualization: Using vision to think*. Morgan Kaufmann, San Francisco, CA.

[5] Carpendale, S. 2008. Evaluating Information Visualizations, in *Information Visualization: Human-Centered Issues and Perspectives*, A. Kerren, J. Stasko, J.-D. Fekete, C. North, Eds. Springer LNCS, Berlin, Heidelberg, Germany, 19-45.

[6] Chang, R., Ziemkiewicz, C., Green, T.M., and Ribarsky, W. 2009. Defining Insight for Visual Analytics. *IEEE Computer Graphics & Applications*, 29, 2 (March/April 2009), 14–17.

[7] Chen, C. and Yu, Y. 2000. Empirical studies of information visualization: a meta-analysis. *International Journal of Human-Computer Studies*. 53, 5 (November 2000), 851-866.

[8] Elmqvist, N., Vande Moere, A., Jetter, H-C., Cernea, D., Reiterer, H., and Jankun-Kelly, TJ. 2011. Fluid interaction for information visualization. *Information Visualization*, 10, 4 (October 2011), 327-340.

[9] Fekete, J.-D., van Wijk, J., Stasko, J., and North, C. 2008. The value of information visualization. In *Information Visualization: Human-Centered Issues and Perspectives,* A. Kerren, J. Stasko, J.-D. Fekete, and C. North, Eds. Springer LNCS, Berlin, Heidelberg, Germany, 1-18.

[10] Heer, J. and Shneiderman, B. 2012. Interactive dynamics for visual analysis. *Communications of the ACM*. 55, 4 (April 2012), 45-54.

[11] Isenberg, P. and Isenberg, T. 2013. Visualization on Interactive Surfaces: A Research Overview. *i-com*, 12, 3 (November 2013). 10-17.

[12] Isenberg, T., Isenberg, P., Chen, J., Sedlmair, M., and Möller, T. 2013. A Systematic Review on the Practice of Evaluating Visualization. *IEEE Transactions on Visualization and Computer Graphics*. 19, 12 (December 2013), 2818-2827.

[13] Keim, D. 2002. Information Visualization and Visual Data Mining. *IEEE Transactions on Visualization and Computer Graphics*. 8, 1 (January-March 2002), 1-8.

[14] Kobsa, A. 2001. An Empirical Comparison of Three Commercial Information Visualization Systems. In *Proceedings of the IEEE Information Visualization Symposium* (October 2001). InfoVis ’01. 123-130.

[15] Lam, H., Bertini, E., Isenberg, P., Plaisant, C., and Carpendale, S. 2012. Empirical Studies in Information Visualization: Seven Scenarios. *IEEE Transactions on Visualization and Computer Graphics*. 18, 9 (September 2012), 1520–1536.

[16] Larkin, J. and Simon, H.A. 1987. Why a Diagram is (Sometimes) Worth Ten Thousand Words. *Cognitive Science*. 11, 1 (January-March 1987), 65-99.

[17] Lee, B., Isenberg, P., Riche, N.H. and Carpendale, S. 2012. Beyond Mouse and Keyboard: Expanding Design Considerations for Information Visualization Interactions. *IEEE Transactions on Visualization and Computer Graphics*.18, 12 (December 2012), 2689–2698.

[18] Norman, D.A. 1993. *Things That Make Us Smart: Defending Human Attributes in the Age of the Machine*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA.

[19] North, C. 2006. Toward Measuring Visualization Insight. *IEEE Computer Graphics & Applications*. 26, 3 (May-June 2006), 6–9.

[20] Pike, W.A., Stasko, J., Chang, R., and O'Connell, T.A. 2009. The Science of Interaction. *Information Visualization*. 8, 4 (Winter 2009), 263-274.

[21] Plaisant, C. 2004. The challenge of information visualization evaluation, In *Proceedings of the International Conference on Advanced Visual Interfaces* (Gallipoli, Italy, May 2004). AVI ’04. 109-116.

[22] Rzeszotarski, J.M. and Kittur, A. 2014. Kinetica: naturalistic multi-touch data visualization. In *Proceedings of ACM Conference on Human Factors in Computing Systems* (Toronto, Canada, May 2014). CHI ’14. 897-906.

[23] Sadana, R., Major, T., Dove, A., and Stasko, J. 2014. OnSet: A Visualization Technique for Large-Scale Binary Set Data. *IEEE Transactions on Visualization and Computer Graphics*. 20, 12 (December 2014), to appear.

[24] Sadana, R. and Stasko, J. 2014. Designing and Implementing an Interactive Scatterplot Visualization for a Tablet Computer, In *Proceedings of the International Conference on Advanced Visual Interfaces* (Como, Italy, May 2014), AVI ’14. 265-272.

[25] Saraiya, P., North, C., and Duca, K. 2005. An Insight-Based Methodology for Evaluating Bioinformatics Visualizations. *IEEE Transactions on Visualization and Computer Graphics*. 11, 4 (July/August 2005), 443-456.

[26] Scholtz, J. 2006. Beyond usability: Evaluation aspects of visual analytic environments. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology.* (Baltimore, MD, October 2006). VAST ’06. 145-150.

[27] Shneiderman, B. 1996. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *Proceedings of the IEEE Symposium on Visual Languages* (September 1996). VL ’96. 336-343.

[28] Shneiderman, B. and Plaisant, C. 2006. Strategies for evaluating information visualization tools: multi-dimensional in-depth long-term case studies, In *Proceedings of the BELIV Workshop* (May 2006). BELIV ’06. 1-7.

[29] Stasko, J., Catrambone, R., Guzdial, M., and McDonald, K. 2000. An Evaluation of Space-Filling Information Visualizations for Depicting Hierarchical Structures. *International Journal of Human-Computer Studies.* 53, 5 (November 2000), 663-694.

[30] Stasko, J., Choo, J., Han, Y., Hu, M., Pileggi, H., Sadana, R., Stolper, C.D. 2013. CiteVis: Exploring Conference Paper Citation Data Visually, (Poster). IEEE Information Visualization Conference (Atlanta, GA, October 2013).

[31] Valiati, E., Pimenta, M., Freitas, C. 2006. A Taxonomy of Tasks for Guiding the Evaluation of Multidimensional Visualizations, In *Proceedings of the BELIV Workshop* (Venice, Italy, May 2006). BELIV ’06. 1-6.

[32] Vande Moere, A. and Purchase, H. 2011. On the Role of Design in Information Visualization. *Information Visualization*. 10, 4 (October 2011), 356-371.

[33] van Wijk, J.J. 2005. The value of visualization. In *Proceedings of IEEE Visualization*, (Minneapolis, MN, October 2005). Vis ’05. 79–86.

[34] Wattenberg, M. 1999. Visualizing the Stock Market, In *Proceedings of ACM CHI Extended Abstracts* (May 1989). CHI ’99 EA. 188-189.

[35] Wehrend, S. and Lewis, C. 1990. A problem-oriented classification of visualization techniques, In *Proceedings of IEEE Visualization* (San Francisco, CA, October 1990), Vis ’90. 139-143.

[36] Yi, J.S., Melton, R., Stasko, J., and Jacko, J. 2005. Dust & Magnet: Multivariate Information Visualization using a Magnet Metaphor. *Information Visualization*. 4, 4 (Winter 2005), 239-256.

[37] Yi, J.S., Kang, Y., Stasko, J.T., and Jacko, J.A. 2007. Toward a Deeper Understanding of the Role of Interaction in Information Visualization. *IEEE Transactions on Visualization and Computer Graphics*. 13, 6 (November/December 2007), 1224-1231.

[38] Yi, J.S., Kang, Y., Stasko, J.T., and Jacko, J.A. 2008. Understanding and Characterizing Insights: How Do People Gain Insights Using Information Visualization. In *Proceedings of the BELIV Workshop* (Florence, Italy, April 2008).

Construction and First Analysis of a Corpus for the Evaluation and Training of Microblog/Twitter Geoparsers

Jan Oliver Wallgrün
GeoVISTA Center
Pennsylvania State University
University Park, PA 16802,
USA
wallgrun@psu.edu

Frank Hardisty
Department of Geography,
GeoVISTA Center
Pennsylvania State University
University Park, PA 16802,
USA
hardisty@psu.edu

Alan M. MacEachren
Department of Geography,
GeoVISTA Center
Pennsylvania State University
University Park, PA 16802,
USA
maceachren@psu.edu

Morteza Karimzadeh
Department of Geography,
GeoVISTA Center
Pennsylvania State University
University Park, PA 16802,
USA
karimzadeh@psu.edu

Yiting Ju
Department of Geography,
GeoVISTA Center
Pennsylvania State University
University Park, PA 16802,
USA
yvj5048@psu.edu

Scott Pezanowski
Department of Geography,
GeoVISTA Center
Pennsylvania State University
University Park, PA 16802,
USA
scottpez@psu.edu

ABSTRACT

This article presents an approach to place reference corpus building and application of the approach to a Geo-Microblog Corpus that will foster research and development in the areas of microblog/twitter geoparsing and geographic information retrieval. Our corpus currently consists of 6000 tweets with identified and georeferenced place names. 30% of the tweets contain at least one place name. The corpus is intended to support the evaluation, comparison, and training of geoparsers. We introduce our corpus building framework, which is developed to be generally applicable beyond microblogs, and explain how we use crowdsourcing and geovisual analytics technology to support the construction of relatively large corpora. We then report on the corpus building work and present an analysis of causes of disagreement between the lay persons performing place identification in our crowdsourcing approach.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Experimentation, Human Factors, Languages

Keywords

geoparsing, corpus building, microblogs, Twitter

SIGSPATIAL'14, November 04-07 2014, Dallas/Fort Worth, TX, USA

http://dx.doi.org/10.1145/2675354.2675701

1. INTRODUCTION

Geographical information is increasingly available, generated by a wide range of GPS enabled devices and other location-based sensors. Since Twitter added the capability for users to attach a location to their tweets, there has been substantial research attention to leveraging that information to support applications such as situational awareness for crisis events [1], mapping social activities related to elections and other events [13], and many other topics that people can be expected to tweet about. However, there is a potentially larger source of place-related information that often goes unused: information about place embedded in text messages and documents. The focus on geolocated tweets, for instance, misses a much larger source of place-relevant information: Only about 1-2% of tweets include geolocation [8], while 10% or more include references to places in the text of the tweet [11].

In more than a decade of research, the field of Geographic Information Retrieval (GIR) has focused on both search for documents that reference or are about places [6]. This research is complemented by work on place name entity recognition and extraction [5, 15] and more recent efforts to extend geoparsing methods to microblog and other more cryptic social media posts [3]. Nowadays a number of geoparsers and related software tools and services exist for identifying and geolocating places in different types of unstructured text ([4], see also [2] for a more exhaustive list), and in microblog or Twitter messages in particular [3, 7]. A key obstacle for progress in GIR in general and the recognition and disambiguation of place references in particular, however, is the current lack of adequate publically available ground-truth corpora of geoparsed text to facilitate the training, evaluation, and comparison of available computational methods and geoparsing systems. While the testing of toponym resolution algorithms based on corpora of news stories has been discussed in the literature [9, 10], the work reported on in this article is to our knowledge the first attempt to address this gap with a focus on microblogs and the goal of providing a publically available benchmarking corpus. The importance

of microblog data has increased drastically in the recent past in many different fields such as science, policy, and humanitarian relief. Microblog data also provides new challenges for geoparsing tools because geoparsing microblogs is harder than geoparsing news stories or other text documents due to the lack of direct context, the presence of abbreviations, special language and notations, etc. (see [3]).

In the research presented in this article, we are developing strategies and tools for geospatial corpus construction and then using the tools to build a ground-truth Geo-Microblog Corpus. It is one of our main aims to generate a corpus that will be freely available to the research community in the near future. In our corpus building work, we adopt a geovisual analytics approach to building a Geo-Microblog Corpus. The approach leverages crowdsourcing strategies through an Amazon Mechanical Turk visual interface to annotate place references in event-based microblog posts, and a visual-computational method to facilitate geographic disambiguation (toponym resolution) and geolocation. The corpus in its current form is based on a selection of 6000 tweets of which 30% contain at least one place name. Occurring place names are marked and annotated with a unique ID and geographic coordinates. We foresee the following applications of our corpus for researchers and developers working on (microblog) geoparsing approaches:

- **Training:** Many geoparsing approaches employ methods that can be improved by training them with a suitable data set. Since place names occurring in our corpus are identified, it can for instance be used to train Natural Entity Recognition (NER) or similar tools.
- **Evaluation:** The corpus provides researchers with a tool to evaluate the performance of algorithms, components, or geoparsing systems in their entirety.
- **Comparison:** Standard benchmarking data sets have significantly spurred research in other fields as they provide a way for quantitatively measuring and comparing the performance of existing systems.

Beyond these applications, such a corpus has applications outside of the main aim of ultimately benefiting the future development of geoparsing tools. It is also a valuable source for more linguistic and cognitive-oriented studies, for instance for analyzing and gaining an understanding of how people conceptualize place and place names, or for comparing the results and performance of experts to that of crowdsourced lay persons. In our analysis of the crowdsourcing results, we encountered interesting causes of disagreement that we expect to have a significant impact on our future corpus building work.

In the remainder of this article, we focus on the underlying corpus building framework and the design decisions made in its development (Section 2). We provide a first analysis of the crowdsourcing-based place identification component of our framework, looking at reasons for disagreement between the persons performing the place identification for the same tweet (Section 3) by introducing our own classification scheme. We close with conclusions and insights gained as well as an outlook on future work in Section 4.

2. CORPUS BUILDING FRAMEWORK

The aim of the framework presented in this section is to build a corpus of tweets in which all occurrences of place names have been identified, and each such place name is annotated with (a) a unique ID to identify it, and (b) its coordinates. For the unique ID, it has been decided to adopt the IDs used by the online gazetteer Geonames[1] (a data source that is often used by geoparsing tools). Coordinates are given in terms of latitude and longitude in WGS 1984. Because of our approach for geocoding the place names (see Section 2.4), these coordinates also stem from Geonames.org and represent the centroids of the respective places. For instance, in the tweet *"Where are the best #steakhouses in New York City? Find out: http://t.co/KJLKJL #nyc"* that refers to New York City twice, but in different forms, each occurrence would be annotated with the information: (1) toponym: New York City; (2) geonamesID: 5128581; (3) latitude: 40.71427; and (4) longitude: -74.00597.

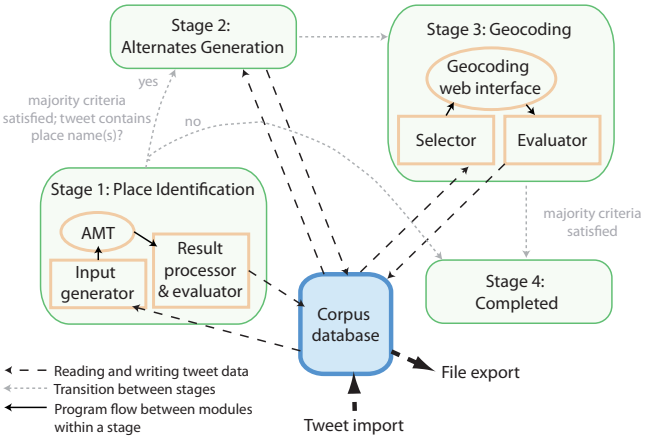


Figure 1: Architecture: In stage 1, place names in the tweet are tagged by AMT workers. In stage 2, for each identified place name a list of potential toponyms is created automatically. In stage 3, a visual interface is used to geocode the place names by selecting the right toponyms from the lists. With reaching stage 4, the ground-truth geoparsing result for a tweet has been completely determined.

To build up such a corpus, we have designed a framework that employs a crowdsourcing approach and different visual interfaces for the two main steps: (a) place name identification and (b) geocoding. The architecture of the framework is shown in Figure 1. A central database is used to store the tweets in the corpus with all required additional information. Each tweet goes through different stages (place identification, alternate generation, geocoding, processing completed). A status field in the database is used to keep track of the processing stage that a tweet currently is at. In the following, we describe the main components including the three main phases (stages 1-3) of corpus generation.

2.1 Tweet Collection and Sampling

Twitter provides an Application Programming Interface (API) through which developers can access and collect tweets. The API used limits the number of tweets accessible, and the textual content of the returned tweets must contain one of the search terms provided with the query. This API has been used to collect close to one billion tweets since January

[1]http://www.geonames.org

1, 2014. The search terms used are selected based upon relatively current events, mainly related to crisis events. The tweets are stored in a PostgreSQL database. From this collection, a random sample of 6000 tweets was drawn by querying for particular keywords: 500 tweets were drawn for each of the 12 event-related terms dengue, malaria, earthquake, measles, fire, protest, flood, rebels, flu, riot, gun, tornado. These keywords were selected from three kinds of events, representing (a) natural disasters, (b) infectious disease, and (c) human threats/violence, that we expected to have a higher number of tweets that are place focused than in an "average" sample (e.g., those about celebrities). Since our corpus is for training and evaluating geoparsers, we were intentionally looking to achieve a higher number of place names and wanted to use our workers efficiently.

When the tweets were initially drawn from the database, many tweets were written primarily in languages other than English. Since our corpus is intended to focus on English tweets, we include the word "the"—the most common word in English and, thus, very unlikely to create any sort of bias—when drawing the tweets. This resulted in only a few exceptions that are not purely in English. All 6000 tweets were imported into the corpus database, making them available for stage 1, the place identification step.

2.2 Crowdsourcing-based Place Identification

To be able to build a large corpus, we are employing a crowdsourcing approach that makes use of Amazon's Mechanical Turk (AMT) platform[2] for the place identification phase. AMT allows for distributing small tasks, referred to as Human Intelligence Tasks (HITs), over a large base of workers. Recruitment and payment is handled efficiently via the AMT platform such that a large number of results can be collected in a short amount of time.

Over the past 5-6 years, crowdsourcing has been used increasingly to annotate corpora and for corpus building in general. Potthast [12] reports on an AMT based project to create a corpus of PAN Wikipedia vandalism edits. In this project, each edit was judged as vandalism or not by three annotators. When 2/3 or more annotators did not agree, the edit was re-annotated by 3 more, and again until disagreements were resolved or until the number that remained undecided was small. These were then resolved manually by the corpus creators. In our approach we adopt best practices established in that article.

To use AMT for our purposes, we designed a web interface allowing an AMT worker to annotate the place names in a tweet. A worker who accepts one of our HITs is presented with a web page with detailed instructions and two tweets that she/he is supposed to tag. Figure 2 shows the central part of the web interface in which a tweet has to be annotated, before and after the annotating. In this interface, every tweet is split into words, each of which is treated as a token. Workers can easily select the token that they recognize as a place name by clicking on that token, which will be highlighted afterwards. As some place names are composed of multiple words/tokens, workers are required to merge these. Our GUI follows a direct manipulation style utilizing HTML5's Drag and Drop feature, which allows the workers to click and drag the text directly, rather than using radio buttons or checkboxes, which are the default UI elements available in the AMT environment. It reduces workload and increases workers' efficiency and productivity. As a result, we can retrieve more results of better quality. In addition to reducing workers' workload, using this interface can also avoid some common identification issues such as misspelling, as workers do not have to type. As an example, for the place name "New York", workers only need to click on the "New", drag it , and drop it onto the "York". Then these two words are merged, and "New York" is formed as one token. To undo the merging, workers just need to double-click on the token, and it will again be split into tokens.

In the instruction part of the web page that workers are presented with, we show an animated image to illustrate the highlighting and merging process. We also provide instructions (with examples) on what should be tagged and what not. Things that should be tagged are: (1) any named town, city, county, state, or country (e.g., Los Angeles, Jefferson County, NY, Italy); (2) named buildings (e.g., Eiffel Tower, Dodgers Stadium, Alcatraz, James J. Ferris High School); (3) named areas (e.g., Grand Canyon, Pacific Ocean, Washington Mall, Hyde Park); (4) street and highway names (Atherton Street, 1st Ave, Highway 1, I-70); and (5) place names inside hashtags (#newyork).

Things not to be tagged are: (1) businesses (Starbucks, Microsoft, Texas Steak House, Baltimore Ravens); (2) organizations (Lutheran Church, Red Cross, United Nations, Grand Canyon Historical Society); and (3) place names used as descriptors (U.S. dollar, Philadelphia cheesesteak).

We pay AMT workers 4 cents per assignment resulting in overall costs of $600 to collect 5 AMT results for each of the 6000 tweets. In addition to giving detailed instructions, we utilize the options provided by AMT to ensure a high quality of the results by only allowing for workers with a HIT acceptance rate of 95% or higher from previous HITs at the start, later increased to 97%. Moreover, each tweet is tagged by at least 5 different AMT workers allowing us to compare the results as further explained below.

HITs are uploaded to AMT in batches. A module in our framework (see again Figure 1) called Input Generator is responsible for selecting the tweets for the next batch based on their status which in turn is based on whether enough results have been collected for this tweet and whether there is a sufficient agreement between the results we get from the different workers. The newly created batch is then uploaded to AMT. Once results for all HITs in a batch have been submitted by the AMT workers, a second software module, the Result Processor & Evaluator, processes the result file downloaded from AMT: Each HIT result is stored in the central database and for each tweet in the batch, all results collected up to this point are compared. We then follow an approach similar to what we described about the work in [12] to decide whether the results satisfy our agreement criteria. We require the following two agreement criteria: (1) ≥ 5 results from different workers exist for this tweet; (2) for each word in the tweet, there is at least a 70% majority agreement on whether that word is part of a place name and if so together with which other words.

If the criteria are satisfied, a ground-truth annotation reflecting the majority vote for each place name/token over all results is created and stored. Stage 1 for this tweet is then finished and its status changed accordingly. If the ground-truth place identification solution for a tweet does not contain any place names, processing of this tweet is completed. Else, the tweet is moved on to stage 2 (alternate generation).

[2]http://www.mturk.com

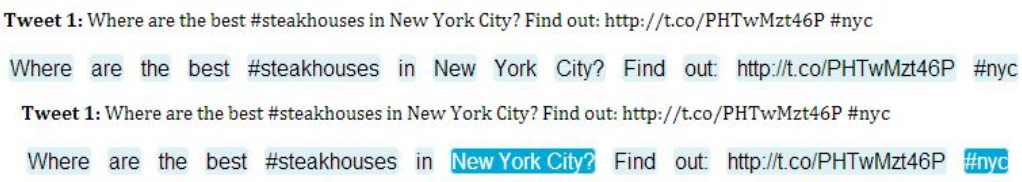


Figure 2: Place identification in the AMT web interface. (a): Tweet before the annotating. (b): Tweet after place names have been highlighted and words belonging to the same place name have been merged.

2.3 Alternates Generation

Since in the geocoding stage (see next section), coders will be allowed to select the correct toponym for each place name in a tweet on a map interface, we need to generate a list of alternative toponyms for each place name, each with its Geonames ID and WGS 1984 coordinates. To generate this list, we use the relevance ranking and geocoding component from our own geoparser GeoTxt [7]. Essentially, this component queries a local Solr index originally created from a Geonames data dump and finds the 100 most likely toponyms for a given place name using our ranking algorithm based on name similarity, population and geographic prominence. The lists are ordered in decreasing order of likelihood. Once these lists have been generated for all place names in a tweet, they are stored with that tweet and the tweet is moved to the geocoding stage.

2.4 Geocoding

In corpus generation, geocoding is the process of annotating the previously identified place names in text with the corresponding geographic coordinates of those places. We use the alternative toponym lists created in stage 2 and an interactive map interface (see Figure 3) to facilitate the lookup process for geocoding. An advantage of using Geonames as the gazetteer for looking up the alternative toponyms is that it is used by many geoparsing tools and the Geonames ID is becoming the defacto standard for linking toponyms in other databases such as DBpedia[3]. However, it has to be noted that the Geonames data is not static. Changes made over time may concern the annotations in our corpus such that a suitable strategy will have to be devised to ensure that the published corpus will remain applicable.

For this stage of corpus building, we use experts, currently PhD students or university faculty members in geography. We have three reasons for this choice: (1) only tweets that have a place name in them (30% of the initial sample) will reach this stage, thus making this approach using individuals with geographic expertise feasible; (2) geocoding place names requires either local knowledge of the place [20] or general geographical knowledge and the willingness to spend extra time to research the existence and location of such a place and its most likely toponym candidate, typically using different web sources; (3) in case of disagreement in geocoding between annotators, we want them to be able to communicate and collectively make decisions, rather than assigning the same task to a new person over and over again without establishing a fundamental understanding of the reasons for disagreement. The communication among expert geocoders has the potential to provide insights on the challenges of geographic disambiguation, including kinds of ambiguity that are common in microblog (or other text) references to place. Therefore, we believe experts are a better choice for the geocoding stage compared to a wider crowdsourcing approach, at least until a set of rules and instructions are generated out of experts' experience with geocoding that would serve as guidelines for the more general public to perform geocoding as well. Once such a rule base is established, a set of well-formed instructions for crowdsourcing has the potential to be used to build larger corpora, a goal that we are pursuing in future research.

For each tweet, the interactive map component of the user interface displays all the places mentioned in the tweet, with the automatically determined toponym with the highest relevance as the default choice for each place name (Figure 3 top). If the automatically assigned toponym is not correct, the annotator is able to click on any of the toponyms to see the alternative candidates for that place name and then pick the correct one (Figure 3 bottom). If none of the presented toponyms is the correct one, this can be indicated by the annotator as well.

Because we are entrusting geocoding to experts, we expect the results of geocoding to be precise. However, to eliminate human error as much as possible, each tweet is assigned to two experts. If their annotation is in agreement, then the resulting annotated tweet will be recorded as a corpus artifact. If there is a disagreement between the two annotations, the tweet will be sent back to both of the annotators to see if they have made a mistake. If both annotators insist on annotations that do not match, that tweet will be flagged and saved for future investigation and guideline establishment.

Using the framework described in this section, we have successfully finished the place identification stage (stage 2) for the 6000 tweets. Since the geocoding part (stage 3) is still in progress, we concentrate our analysis in the next section on the results of the AMT-based place identification stage.

3. A FIRST ANALYSIS OF THE CORPUS

In this section, we provide a first analysis of the corpus collected, focusing on the results from the place identification stage. We are considering the questions of how often place names occur in our corpus and what the sources of disagreement between AMT workers in the place identification stage are. After proposing a classification scheme for the sources of disagreement, we discuss what we can learn from these disagreements about place conceptualization generally and more specifically, how we can improve the components of our corpus building framework and resolve different kinds of disagreement to make the results usable for our corpus.

3.1 Frequency of Place Names

One of the objectives when starting to create the corpus

[3]http://dbpedia.org

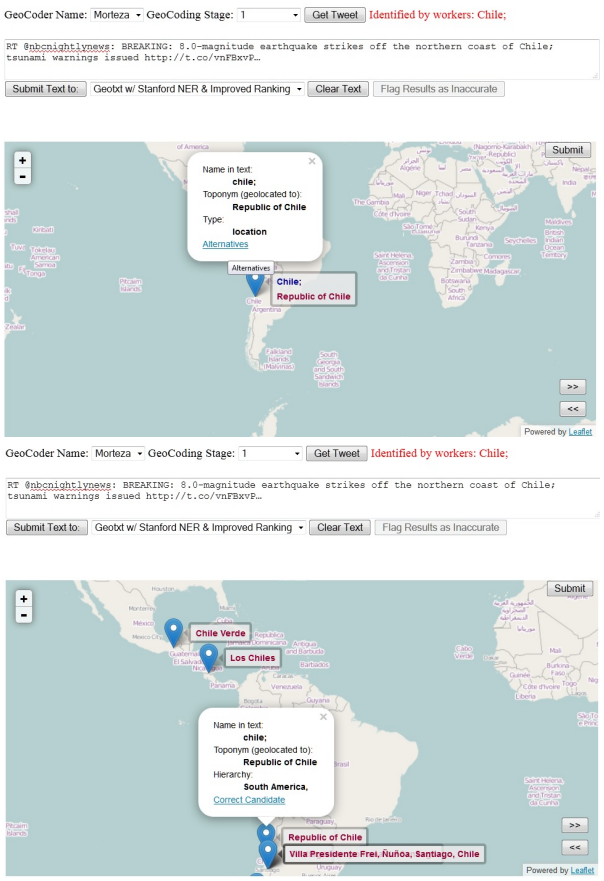


Figure 3: Interactive geocoding interface. Top: Initial situation with automatically assigned toponyms (here 'Chile"). Bottom: Alternative toponyms are displayed allowing the expert to pick the right one.

was to achieve a higher frequency of place names than in a completely random sample. Overall, it turns out that of the event-based sample of tweets making up our corpus 70.0% contain no place names. 22.3% contain one place name. 5.7% contain two place names and 1.3% more than two place names. Thus, the percentage with place names is much higher than the 10% tweets with place names that we obtained in previous work by querying the twitter API for about 150 terms that had greater diversity.

3.2 Place Identification Disagreement

While our corpus collection framework automatically creates more HITs for tweets for which the agreement criteria in the place identification phase have not been satisfied, we blocked further processing of tweets where results did not seem to converge after getting more results than the initial five. With this threshold, agreement for place identification has been reached for 91.6% of the 6000 tweets in our corpus.

Disagreement between the workers can have many reasons and as argued by Wong and Lee [14] is potentially due to ambiguity in natural language, thus recording situations of disagreement can provide important information about that ambiguity. Wong and Lee report on a project directed to dealing with annotator disagreement in corpus construction. Specifically, they outline current approaches to dealing with disagreement that include: providing detailed annotation guidelines (thus making the annotators more expert), using expert adjudication (by a subject matter expert), discussion among annotators, removal of entities that agreement cannot be reached on, relaxing criteria that allow slightly different judgments to be counted as the same, and crowd wisdom (often used with AMT, in which it is possible to add annotators until some predetermined level of agreement is reached). With this background, Wong and Lee argue that the approach in most corpus building to remove all disagreement and keep only entities that sufficient agreement has been reached on is flawed as it fails to consider "legitimate" disagreement due to ambiguity in natural language

In the following, we take a look at the reasons for disagreement in place identification by establishing a novel classification scheme. After inspecting the tweets, we came up with the following categories of causes of disagreement between AMT workers, grouped into three main classes: tweeter errors, worker errors, and the most interesting category of disagreement based on differing definitions of place name.

3.2.1 *Tweeter Errors (TWE)*

This category refers to cases in which the disagreement is rooted in typos, etc. in the original tweet leading to uncertainty on the side of the AMT workers on whether something constitutes a place name that should be marked or not.

Example: *"Its been 20 years since the north ridge earthquake? Wow time flies"*

In this example the tweeter erroneously added a space in the name Northridge (and did not capitalize it). We believe that this is the reason why several workers did not tag it.

3.2.2 *Worker Errors*

This category refers to cases where workers make mistakes that clearly were against the instructions given to them. We distinguish the following subgroups:

Familiarity / unfamiliarity (FAM): In some cases, it was clear that some less well-known places were not recognized by some of the workers, resulting in disagreement.

Example: *"... help us find missing white large dog Leechan http://t.co/pxZY5wmpov Sendai #Japan. one of ..."*

We believe that those AMT workers who did not highlight Sendai in this tweet were not familiar with that name.

Misidentified tokens (MTN): Workers sometimes identified tokens as place names when the tweeter was not intending to refer to a place name.

Example: *"@lakeline @HockeenightsCT great. Now youve done it CT. Youve ..."*

The facts that "CT" appears in a user handle and the "Now youve done it CT" make it clear that CT refers to a person's name, rather than the state of Connecticut in the U.S. In the context of tweets it can sometimes be tricky or even impossible to decide about such a case.

Non-Merging errors (NME): Despite our instructions and the animated examples on the web site, several workers did not get the idea of having to merge the words making up a single place name or were unable to do so.

Example: *"... Earthquake Shakes Mexicos Pacific Coast: A strong earthquake has shaken the southern Pacific... "*

In this tweet, we had two workers that highlighted "Mexicos Pacific Coast" and "southern Pacific" but without any

word merging; each word was tagged as a place name by itself. This makes it pretty clear that they were not aware that they had to or not able to merge the words.

Skipped repeats (REP): We found cases in which the same place appears several times in the same tweet and some workers only marked one of these occurrences. Since we instructed the workers to "mark all place names", we feel that these cases belong into the worker errors category.

Example: *"... Text: No measles outbreak in Cavite - TRECE MARTIRES CITY, Cavite There is still no ..."*

In this example, some workers only highlighted either the first or second of the two occurrences of "Cavite".

3.2.3 Disagreement Based on Differing Definitions

This category refers to cases where reasonable people may disagree about what words refer to a place. They are of special interest for both practical and theoretical reasons. Practically, they contain the most common sources of disagreement, and so we need to understand them to maximise our yield of identifications for our corpus. Theoretically, examining these sources of disagreement can give us insight into how people refer to and recognize places, and help us make progress on the overall question, "what is a place?"

Adjectival usage (ADJ): A place name is used in an adjectival way to describe a different object rather than to refer to a place by itself.

Example: *"S'Sudan rebels reinforce captured city defences: South Sudan's rebels are strengthening ..."*

"South Sudan's rebels" is not a place and neither "S'Sudan", the way it is used here. In this tweet, the place name is used as an adjective to modify an object. While our instructions say that "Place names used as descriptors" should not be tagged, some workers did not take this into account.

Organization with a place in name (ORG): In these cases, a place name does not explicitly refer to a place but instead is part of the name of an organization or business.

Example: *"#Patriots How the Indianapolis Colts Defense Can Keep Tom Brady in Check http://t.co/TxBtIhHlSS"*

In this example, some workers highlighted Indianapolis despite our instructions saying that places in organization or business names should not be tagged including the example "Baltimore Ravens" which is very similar to this one.

Place hierarchy related issues (PHR): In our set of tweets with high disagreement, we find quite a few examples in which a place name further up in the place hierarchy (parent) is used to unambiguously identify another name (child). Typically, the place names appear separated by a comma, for instance <city>, <state> (e.g. "Hartford, CT").

Example: *"RT @NewEarthquake: 5.6 earthquake, 297km ENE of Grytviken, South Georgia and the ..."*

In the submissions of the AMT workers we find several versions of how such a case is handled. Some highlight both places, child and parent, both individually meaning they are not merged together. Some highlight both and merge them together presumably under the logic that it is actually one place that is referred to here. We also find workers that only highlight the child (Grytviken) most likely also thinking that this is the place referred to and the parent only occurs to clearly identify it. Lastly, some workers also only highlighted the parent (South Georgia). The reasoning behind this is less clear; it is possible that these are simply cases of unfamiliarity with the place name of the child (FAM).

Non-proper name (NPN): Places are often referred to without a proper place name in the designation. This results in these references being marked by only part of the workers.

Example: *"... Flash flood watch has now been trimmed back to just include the south shore through 9am."*

Here "south shore", while referring to a particular location, does not contain a proper name. Typically, in such situations it is extremely challenging from a natural language interpretation perspective to decide what specific place is referred to. Without more context, the reference refers to a class of places rather than a particular place.

Vaguely qualified place name (VQP): This describes a situation where there is a proper place name with an addition in front of it that narrows things down spatially, but at the same time introduces some vagueness.

Example 1: *"... wildfires in south #California: At least two structures burned to the ground an..."*

Example 2: *"... Theres an outbreak of measles in the Bronx and upper Manhattan, officials warn: ..."*

The qualifiers ("south" and "upper") narrow things down spatially to a smaller area within the spatial extension of the properly named place. It also becomes less clear which exact area is being referred to. Workers either consider the place name to consist of the complete construct ("south #California" in the first example, "upper Manhattan" in the second) or highlight just the proper name without the qualifier.

Inclusion or exclusion of kind of place (KND): We often find the name of a place in a combination with a noun describing its kind, such as in City of London, Island of Saint Louis, South Georgia Island, Ganges River, etc.

Example: *"... How Adelia, 9, and the island of Sabang, #Indonesia, triumphed over #malaria ..."*

Part of the AMT workers here highlight and combine "island of Sabang" and the other part only Sabang. This reflects the uncertainty on whether the kind noun is part of the place name or not. For many physical features such as lakes, mountains, etc., it is clear that the noun has to be considered part of the proper name. This is particularly important from a geoparsing perspective where otherwise it would be impossible to distinguish the physical feature from a similarly named populated place, e.g. Lake Michigan vs. the state Michigan. For other entities such as cities, villages, or states, it is often difficult to make a decision even for geographic experts, and it requires some background research.

Hashtag related issues (HAS): While we give clear instruction that "Place names in hashtags" should be tagged using the example #newyork, it turns out that there are many more cases not clearly covered by these instructions where the place name is only part of the hashtag.

Example: *"... everyone please pray for the people there. #prayforindonesia"*

Since in our current tool there is no way that would allow workers to highlight part of a token, some workers highlighted the hashtag #prayforindonesia because of the presence of Indonesia, while others did not highlight it. In addition to that, we can find in hashtags many of the other cases like organization names containing place names in the hashtag (ORG), adjectival usage (ADJ), etc. In principle, this category can occur in combination with any of the others.

Below we summarize for each category roughly how many of the contentious tweets had submissions falling into that category (several categories per tweet are possible).

TWE	FAM	MTN	NME	REP	ADJ
1%	32%	5%	20%	6%	23%
ORG	PHR	NPN	VQP	KND	HAS
16%	11%	5%	10%	6%	16%

3.3 Resolving the Disagreements

Resolving the disagreements of the AMT workers to proceed with the geocoding step and include these tweets in the corpus is crucial for avoiding an undesired bias. Many of these tweets are particularly interesting cases for evaluating or training geoparsing tools as they tend to contain several place names, often in a spatial relationship, e.g., category PHR. There are different possibilities of how to deal with annotator disagreement (see again [14]). We here consider the following three options, whose suitability depends on the disagreement categories introduced previously: (1) manual resolution of the disagreements by experts that decide (ideally collectively) what are correct submissions (according to our own rules) and the ground truth result for the controversial tweets; (2) automatic resolution via algorithms that can correct or discard incorrect submissions (according to our own rules plus the set of entity designations by workers); and (3) introducing new rules to instruct workers more clearly and collect more results via AMT.

These measures can also be combined and probably will have to be. New instructions alone will probably not be sufficient in all cases, as we found errors in the current results that violate some of the instructions already given to the workers. Discarding submissions that are incorrect (according to our own rules), either manually or where possible automatically, can significantly reduce the number of new submissions needed to achieve the specified majority.

Starting with the tweeter errors (TWE), a decision has to be made on whether misspelled place names should count and be highlighted. We believe that including misspelled names is an appropriate strategy because robustness against typing errors is a desirable property in geoparsing tools. On the other hand, these errors are sometimes extremely difficult to detect automatically. In these cases, where autocorrection of spelling is impractical, manual resolution and/or new instructions to AMT workers are the best options.

In the case of worker errors, unfamiliarity reasons (category FAM) and misidentified tokens (category MTN) can obviously not be resolved by improved instructions. Automatic resolution based on a geographic gazetteer is also problematic as this is a form of geoparsing and could lead to a bias in the corpus. Hence, manual resolution by experts seems to be the only viable approach in these cases. As we discussed, we already provide explicit instructions that should prevent non-merging errors (category NME) and skipped repeats (category REP) but we do not completely achieve these goals. On the positive side, these are categories of worker errors where automatic resolution seems to be a promising approach. In the case of NME errors, a sequence of tokens that is merged and highlighted by the majority of workers who processed this tweet and that was highlighted but not merged by the rest can be easily detected and the latter treated as if they were merged.

This leaves the categories based on different definitions of what constitutes a place. For adjectival usage (category ADJ) and for organization with a place in the name (category ORG) we already provide instructions that say that these should not be marked as place names, as it is not the place described by that name that is being referred to in the tweet. The instructions and examples for these cases could potentially be improved but there are hard cases where it is difficult to decide and where simple rules will not always work. Since automatic resolution in these cases is extremely challenging from an algorithmic and natural language processing perspective, manual resolution seems to be the most suitable approach to get results quickly. For place hierarchy issues (category PHR), we have made the decision that we want both place names (child and parent) highlighted separately without merging. Fortunately, automatic resolution techniques are applicable here as these cases are rather easy to detect and resolve. Moreover, we expect that giving explicit instructions about this case to the AMT workers will lead to a substantial reduction in these errors.

The remaining categories have in common that it is not straightforward to decide what the preferred worker behavior is with regard to whether something should be tagged or not. Expressions such as “south shore” or simply “South West” falling into the NPN category refer to a place, but not with a proper place name. In the context of geoparsing, we believe it is more reasonable to restrict oneself to proper place names. We, therefore, do not want these references to be tagged. Unfortunately, detecting NPN cases automatically is extremely challenging. However, we expect that providing clear instructions with examples that tell AMT workers to only highlight proper place names should reduce the frequency in which this case occurs significantly.

Vaguely qualified place names (VQP) such as “south California” or “upper Manhattan” is another case where one can argue about what the best approach is. One can either take the standpoint that geoparsers should essentially process the proper name without the vague qualification, which is extremely difficult to interpret in a geometric sense anyway, or attempt to deal with the entire phrase in a reasonable way, e.g. by indicating or visualizing that the identified proper place has been qualified further. We are currently leaning towards focusing on the proper place name for training and evaluation purposes and treat the processing of additional qualifications as an extra feature of a geoparser whose evaluation is outside of the scope of this corpus. Fortunately, it is straightforward to provide clear instructions on the desired worker behavior in future collections as well as to automatically resolve (most of) these cases. This also opens up the option to introduce a special labeling of these cases distinguishing the proper name and the qualification component such that the corpus could also be used specifically to benchmark this aspect in a geoparser.

As we already briefly discussed, the inclusion or exclusion of kind of place (KND) such as “City of London” or “Lake Michigan” is another difficult case even experts might struggle with. We believe that for physical features (mountains, rivers, etc.) these should be tagged as part of the proper place name but we are less certain about populated places (“city of”, “village of”, etc.) and similar cases. A reasonable approach might be to instruct workers to include them but allow for some flexibility when comparing the output of a geoparser with an instance from the corpus in a benchmark-

ing scenario. It should be possible to resolve the majority of these cases automatically and the same holds true for implementing some special treatment of these for training.

In the case of place names in hashtags (category HAS), it is our opinion that they should be tagged if they contain place names, unless they fall into one of the categories discussed above where we argued these should not be tagged. That means that in cases where several place names are combined in a single hashtag, several places need to be associated with a single token as in #PrayForSerbiaAndBosnia. As a future refinement it would make sense to provide workers with a way to mark which parts of a hashtag constitute a place name. Improving the instructions to cover the case that a hashtag consists of more than just a place name should improve the results we get from the workers. Moreover, it should be possible to detect and resolve at least some of these cases automatically in the current corpus.

4. CONCLUSIONS & OUTLOOK

We have presented our initiative of constructing a GeoMicroblog Corpus that we plan to publish in the near future to facilitate the training, evaluation, and comparison of geoparsing tools and promote GIR research. The details of releasing the corpus are still under discussion due to legal issues and other considerations that need to be taken into account. The corpus building framework we described adopts geovisual analytics and crowdsourcing components and we expect it to play an important role in further corpus building work, for instance for other kinds of text or for building more domain specific or localized corpora.

The analysis of submissions in the place identification stage brought several interesting reasons for disagreement to light. Based on the insights gained, we plan on evaluating three strategies for making tweets with high disagreement useful for our corpus and increasing the yield of usable place name references in future collections. These are manual decision making by experts, improving instructions and collecting more results, and automatic resolution methods.

Furthermore, one of the next steps will be to apply the corpus to quantitatively evaluate the performance of our own geoparsing web service GeoTxt [7]. In addition, we plan to use the results from the place identification stage to train the NER tools in our system. We expect new insights from these activities which should lead to improvements of the corpus and corpus building framework. Finally, we are going to look into ways to actively disseminate the corpus in the research community and discuss options for establishing a benchmarking initiative or competition in the field.

5. ACKNOWLEDGMENTS

We thank the anonymous reviewers for very valuable feedback and suggestions. This work was partially funded by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009- ST-061-CI0003.

6. REFERENCES

[1] A. Crooks, A. Croitoru, A. Stefanidis, and J. Radzikowski. #Earthquake: Twitter as a distributed sensor system. *Transactions in GIS*, 17:124–147, 2012.

[2] C. D'Ignazio, R. Bhargava, E. Zuckerman, and L. Beck. Cliff-clavin: Determining geographic focus for news. In *NewsKDD: Data Science for News Publishing, at KDD 2014*, 2014.

[3] J. Gelernter and S. Balaji. An algorithm for local geoparsing of microtext. *GeoInformatica*, 17(4):635–667, 2013.

[4] R. Guillén. Geoparsing web queries. In *Advances in Multilingual and Multimodal Information Retrieval*, volume 5152, pages 781–785. Springer, 2008.

[5] Y. Hu and L. Ge. A supervised machine learning approach to toponym disambiguation. In *The Geospatial Web – How geobrowsers, social software and the Web 2.0 are shaping the network society*, pages 117–128. Springer, 2007.

[6] C. Jones, R. Purves, A. Ruas, M. Sanderson, M. Sester, M. Kreveld, and R. Weibel. Spatial information retrieval and geographical ontologies. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 387–388, 2002.

[7] M. Karimzadeh, W. Huang, S. Banerjee, J. O. Wallgrün, F. Hardisty, S. Pezanowski, P. Mitra, and A. M. MacEachren. GeoTxt: A web API to leverage place references in text. In C. Jones and R. Purves, editors, *Proceedings of the 7th Workshop on Geographic Information Retrieval*, pages 72–73, 2013.

[8] K. Leetaru, S. Wang, G. Cao, A. Padmanabhan, and E. Shook. Mapping the global Twitter heartbeat: The geography of Twitter. *First Monday*, 18(5), 2013.

[9] J. L. Leidner. An evaluation dataset for the toponym resolution task. In *Computers, Environment and Urban Systems*, volume 30, 2006.

[10] J. L. Leidner. *Toponym Resolution in Text*. PhD thesis, School of Informatics, University of Edinburgh, 2007.

[11] A. MacEachren, A. Jaiswal, A. Robinson, S. Pezanowski, A. Savelyev, P. Mitra, X. Zhang, and J. Blanford. Senseplace2: GeoTwitter analytics support for situational awareness. In S. Miksch and M. Ward, editors, *IEEE Conference on Visual Analytics Science and Technology*, 2011.

[12] M. Potthast. Crowdsourcing a Wikipedia vandalism corpus. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 789–790, 2010.

[13] M.-H. Tsou, J.-A. Yang, D. Lusher, S. Han, B. Spitzberg, J. Gawron, D. Gupta, and L. An. Mapping social activities and concepts with social media (Twitter) and web search engines (Yahoo and Bing). *Cartography and Geographic Information Science*, 40:1–12, 2013.

[14] B. Wong and S. Lee. Annotating legitimate disagreement in corpus construction. In *Sixth International Joint Conference on Natural Language Processing*, 2013.

[15] D. Woodward, J. Witmer, and J. Kalita. A comparison of approaches for geospatial entity extraction from Wikipedia. In *IEEE Fourth International Conference on Semantic Computing (ICSC)*, pages 402–407, 2010.

Place Reference in Text as a Radial Category: A Challenge to Spatial Search, Retrieval, and Geographical Information Extraction from Documents that Contain References to Places

ALAN M. MACEACHREN
Professor,Geography
Affiliate Professor, Information Sciences & Technology
Director, GeoVISTA Center
The Pennsylvania State University
Email: maceachren@psu.edu

The concept of place is fundamental to spatial search as it relates to finding geographical information in heterogeneous information repositories as well as to the kinds of "spatial search" that are central to human wayfinding in the world. This position statement focuses on place in the context of spatial search for and within text documents, but the conceptualization of place proposed has implications for spatial search more broadly. Spatial search in text documents requires solving a range of challenges related to place, including: recognizing place entity mentions in text, disambiguating and locating the place entities, and determining whether the document is "about" the place(s) mentioned or those place mentions serve some other purpose.

Research over the past several years has provided a base from which to address each of these questions (Zhang et al. 2009; Zhang et al. 2010; Jaiswal et al. 2011; Zhang et al. 2012; Karimzadeh et al. 2013; Xu et al. 2014; Wallgrün et al. submitted). Based upon that research, I propose that one of the most fundamental issues in spatial search focused on retrieving and extracting geographically-relevant information from documents is understanding and being able to recognize what constitutes a reference to a place. My argument here is that place reference is best conceptualized as a *radial category*, but that current entity recognition approaches take a classical category theory approach to recognizing place references. This existing approach impedes strategies for spatial search in text as well as those for geoparsing and mapping the content of documents.

It is important to distinguish between "place" and "place reference." Conceptualization of place is also a fundamental research question (Goodchild 2011) and category theory (within which the concept of radial categories is defined) has been applied successfully to investigate conceptualization of place (Mark, Smith, and Tversky 1999; Lloyd, Patton, and Cammack 1996). Here, however, my focus is on linguistic or textual references to place and how to recognize and interpret them. Thus, the focus is not on what a place is conceptually or in practice (although conceptualization of place is important in understanding place references) but on distinguishing references to place from references to people, organizations, and other entities.

Radial categories (Lakoff 1987), are structured with respect to a prototype and include members in relation to some "center" or prototype (Janda 2010). While I am not aware of any empirical research focused on identifying prototype place references (there is work on basic level geographic features that includes place concepts, [e.g., Lloyd, Patton, and Cammack 1996]),

nor the characteristics of "place reference" as a category, I can speculate that London or Paris might constitute an exemplar, thus close to a prototype place reference for many people. But, someone from China might center their conceptualization of place on Beijing while someone from India might have New Delhi as their prototype. On the other hand, an Amish farmer from the eastern U.S. might center their concept of place on the home farm or local livestock market. Thus, the category of place differs for individuals from different places and who have a different pattern of behavior in the world. Whatever the prototype, radial categories include entities that are more or less representative, thus more or less central or peripheral to the prototype. The position in this radial category space has implications for strategies used to recognize place references in text and for how such references are tagged for subsequent use.

For the category of "place references" (in text or speech), centrality (and thus the likelihood that a speaker/writer, in a statement containing a place name or related reference, is specifically thinking of an entity as a "place" versus another entity type) is likely to depend upon a range of factors. As part of recent research to develop tools for place reference corpus building and then to apply the tools to building a Twitter place reference corpus (Wallgrün et al. submitted), several factors were identified as responsible for disagreements among human annotators (given the task of tagging place references in tweets). Drawing on this work and further analysis of our tweet-tagging results, I propose the following as factors that influence whether a named entity was intended as a place reference; the combination of factors will determine relative position of a reference within or outside the radial place reference category:

Geographical scale: Research has demonstrated that bonds people form with places differ according to geographical scale (e.g., neighborhood, city, country) (Bernardo and Palma-Oliveira 2013). The intensity of place identity and attachment is likely to influence the extent to which a named entity is conceptualized as a "place."

Use as a noun versus adjective: Names as nouns are more certainly place references than names used as an adjective. But, use as a possessive adjective is intermediate. Given three endings of the phrase "I'm going to {the Pirates game in Pittsburgh; Pittsburgh's Pirates game; the Pittsburgh Pirates game}", the first is clearly a place reference and the last is clearly a reference to a sports team (that is from a place). The middle case, however, can be interpreted as a reference to the place, which has a sports team for which only the nickname is given.

Precision of reference: The distinction between an organization and an instance of that organization that is a place is a function of precision of reference. The statement "I often drink Starbucks coffee" refers to the company while "We usually stop at Starbucks" might be a reference to a particular coffee shop or to the chain generally. In contrast, "let's meet at the Starbucks in the hotel lobby" is more certainly a reference to a (within building) place.

Interdependence: Entities can be clear place references in one context and a qualifier of a separate place reference in another. For example, in the two version of this sentence: "I'm traveling to {Illinois; Springfield, Illinois}, the first refers to the state as a distinct place while the second uses "Illinois" to clarify which Springfield is intended (with Illinois secondary).

Agency or lack of it: Place names can be used in ways that imply agency; in these cases, there is a tension between conceptualizing the reference as a place versus as an agent exhibiting behavior. An example is: "Canada Takes a Wait-And-See Approach to New Cage Regulations," Alberta Farmer, Posted Nov. 7, 2011 by Sheri Monk in Livestock; accessed Sept. 19, 2014

(http://www.albertafarmexpress.ca/2011/11/07/canada-takes-a-waitandsee-approach-to-new-cage-regulations/).

This statement only scratches the surface of the potential application of cognitive category theory to conceptualization of place in the context of spatial search. There is a range of past research on application of category theory to understanding how geographical features are conceptualized (e.g., Mark, Smith, and Tversky 1999; Usery 1993) and newer work on application to contexts such as tagging Flickr photos (e.g., Stvilia and Jörgensen 2010). A next step is to focus on integrating knowledge about how place and related geographic features are conceptualized with that on how place is referred to in text. This integration has the potential to enhance our ability to search for documents that are "about" places of interest and to recognize, extract, disambiguate, and locate the references. One objective is to develop entity recognition methods that take context into account and that can recognize different kinds and intensities of place reference so that spatial search for geographic information in text documents can be better tailored to particular use contexts.

References:

Bernardo, F., and J. Palma-Oliveira. 2013. Place identity, place attachment and the scale of place: The impact of place salience. *Psyecology* 4(2): 167–193.

Goodchild, M. F. 2011. Formalizing Place in Geographic Information Systems Communities, Neighborhoods, and Health. In *Communities, Neighborhoods, and Health*, eds. L. M. M. Burton, S. A. P. Matthews, M. Leung, S. P. A. Kemp and D. T. T. Takeuchi, 21–33: Springer New York.

Jaiswal, A., X. Zhang, P. Mitra, S. Pezanowski, I. Turton, S. Xu, A. Klippel, and A. M. MacEachren. 2011. GeoCAM: A Geovisual Analytics Workspace to Contextualize and Interpret Statements about Movement. *Journal of Spatial Information Science* 3: 65–101.

Janda, L. A. 2010. Cognitive linguistics in the year 2010. *International Journal of Cognitive Linguistics* 1 (1): 1–30.

Karimzadeh, M., W. Huang, S. Banerjee, J. O. Wallgrün, F. Hardisty, S. Pezanowski, P. Mitra, and A. M. MacEachren. 2013. GeoTxt: A Web API to Leverage Place References in Text In *7th ACM SIGSPATIAL Workshop on Geographic Information Retrieval*. Orlando, FL: ACM.

Lakoff, G. 1987. *Woman, Fire, and Dangerous Things: What Categories Reveal about the Mind*. Chicago: University of Chicago Press.

Lloyd, R., D. Patton, and R. Cammack. 1996. Basic-level geographic categories. *Professional Geographer* 48(2): 181–194.

Mark, D. M., B. Smith, and B. Tversky. 1999. Ontology and Geographic Objects: an empirical study of cognitive categorization. Paper read at COSIT'99—Conference on Spatial Information Theory, at Berlin.

Stvilia, B., and C. Jörgensen. 2010. Member activities and quality of tags in a collection of historical photographs in Flickr. *Journal of the American Society for Information Science and Technology* 61(12): 2477–2489.

Usery, L. 1993. Category theory and the structure of features in geographic information systems. *Cartography and Geographic Information Systems* 20(1): 5–12.

Wallgrün, J. O., M. Karimzadeh, A. M. MacEachren, F. Hardisty, S. Pezanowski, and Y. Ju. submitted. Construction and First Analysis of a Corpus for the Evaluation and Training of Microblog/Twitter Geoparsers.

Xu, S., A. Klippel, A. M. MacEachren, and P. Mitra. 2014. Exploring regional variation in spatial language using spatially-stratified web-sampled route direction documents. *Spatial Cognition and Computation*.

Zhang, X., P. Mitra, A. Jaiswal, S. Xu, A. M. MacEachren, and A. Klippel. 2009. Extracting route directions from webpages. Paper read at Twelfth International Workshop on the Web and Databases (WebDB 2009) June 28, 2009, at Providence, Rhode, Island, USA.

Zhang, X., P. Mitra, A. Klippel, and A. MacEachren. 2010. Automatic Extraction of Destinations, Origins and Route Parts in Human Generated Driving Directions. Paper read at GIScience 2010, at Zurich, Switzerland.

Zhang, X., B. Qiu, S. Xu, P. Mitra, A. Klippel, and A. M. MacEachren. 2012. Disambiguating Road Names in Text Route Descriptions using Exact-All-Hop Shortest Path Algorithm. Paper read at European Conference on Artificial Intelligence (ECAI), Aug. 27–31, at Montpellier, France.

An Intelligent Crowdsourcing System for Forensic Analysis of Surveillance Video

Khalid Tahboub, Neeraj Gadgil, Javier Ribera, Blanca Delgado, and Edward J. Delp

Video and Image Processing Laboratory (VIPER)
School of Electrical and Computer Engineering
Purdue University, West Lafayette, Indiana USA

ABSTRACT

Video surveillance systems are of a great value for public safety. With an exponential increase in the number of cameras, videos obtained from surveillance systems are often archived for forensic purposes. Many automatic methods have been proposed to do video analytics such as anomaly detection and human activity recognition. However, such methods face significant challenges due to object occlusions, shadows and scene illumination changes. In recent years, crowdsourcing has become an effective tool that utilizes human intelligence to perform tasks that are challenging for machines. In this paper, we present an intelligent crowdsourcing system for forensic analysis of surveillance video that includes the video recorded as a part of search and rescue missions and large-scale investigation tasks. We describe a method to enhance crowdsourcing by incorporating human detection, re-identification and tracking. At the core of our system, we use a hierarchal pyramid model to distinguish the crowd members based on their ability, experience and performance record. Our proposed system operates in an autonomous fashion and produces a final output of the crowdsourcing analysis consisting of a set of video segments detailing the events of interest as one storyline.

Keywords: crowdsourcing, video surveillance, video analytics, forensic analysis

1. INTRODUCTION

Video surveillance systems are widely deployed for public safety [1], [2]. In recent years, the number of surveillance cameras and security systems consisting of multiple cameras have grown exponentially. The video contents generated by such systems are often archived for forensic purposes. Search and rescue missions and "after-the-event" investigations are examples of time-critical forensic tasks that require a careful analysis of videos of tens or hundreds of hours of duration. Annually, hundreds of search and rescue missions are carried out in North America and many more worldwide. The search for the lost Malaysian airplane (MH370) and the April 15, 2014 Boston bombing investigation are some examples of such time-critical missions.

In recent years, there has been a great effort by the image/video processing, computer vision and artificial intelligence communities to develop intelligent systems capable of real-time monitoring and alerting [3], [4]. Video analytics have been developed for object detection, tracking and categorization, human action and behavior analysis [5], [6], [7], [8], [9], [10], [11], [12], [13], [14]. However, these methods still face significant challenges such as dealing with occlusions, shadows, illumination changes and the requirements to track objects across multiple cameras. The limitations of automatic systems are highlighted in the failure cases from [15], [16], [6]. Improving the performance of such intelligent systems has become an active research area within the computer vision community. Combining the concepts of machine learning with human computation resources can provide an effective solution to meet the demands of forensic analysis of surveillance video.

Crowdsourcing was defined by J. Howe in 2006 as "taking a function once performed by employees and outsourcing it to an undefined (and generally large) network of people in the form of an open call"[17]. It is also referred to as collective intelligence, the wisdom of the crowd or human computation. Crowdsourcing is often considered as an effective solution to problems that involve cognitive tasks [18]. Crowdsourcing platforms such as Amazon Mechanical Turk (MTurk) [19], Freelancer [20] and Mob4hire [21] aim to use the collective intelligence

This work was partially supported by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0001. Address all correspondence to Edward J. Delp, ace@ecn.purdue.edu

of crowds to do tasks that machines find very difficult. Law enforcement authorities have used some forms of crowdsourcing to reach out to volunteers and ask them to search for objects, suspicious events or missing people in video. The search for the Malaysian airplane had a publicly available crowdsourcing web-based platform. Other search and rescue missions use YouTube and ask volunteers to view videos online.

Describing the important contents and actions in a video sequence is known as video annotation [22]. This is an area in which crowdsourcing can be very effective [23]. The video event representation language (VERL) is a formal language for representing events for designing an ontology for an application domain and for annotating data with the ontology's categories [24]. An example of this is sports annotation as described in [25]. Another area of interest is object detection and tracking. Crowdsourcing-based annotation is very useful not only for obtaining the annotations, but for improving the performance of automatic detection methods. In [26], *MTurk* is used to provide annotations to train object detectors where the system automatically refines its models by actively requesting annotations of images from the crowd. In [27], a system in which machine learning and crowdsourcing enhance each other is proposed. In this system, a semi-automatic image annotation approach is presented that uses crowdsourcing to help robots register novel objects with their semantic meaning. A web-based social analysis tool is proposed in [28] in which several key strategies are presented that improve the quality and diversity of explanations generated by the crowd members. In [29], object classifiers are trained and continuously refined by reaching out to members of the crowd through MTurk. In [30], a system for enhancing machine learning using crowdsourcing is proposed.

A group of people can perform the same or sometimes even better than an expert when the crowd satisfies the basic conditions: diversity, independence, decentralization and aggregation [18]. With this basic intuition, there have been many studies utilizing human intelligence to help machines perform better. In [31], we described a web-based system for the annotation of surveillance video in an organized and controlled way. Crowdsourcing surveillance video is described in [32] where *CrowdFlow*: an *MTurk*-based toolkit for integrating machine learning with crowdsourcing is presented. In [33], a method to enhance automated crowd flow estimation using crowdsourcing is presented. *Crowded* is another such web-based platform developed by the defense science & technology laboratory (DSTL) in which, images of a particular location are collected from a variety of media sources to provide an operator with real-time situational awareness [34]. A similar *MTurk*-based web interface, known as *VATIC* was developed to monetize, high quality crowdsourced video labeling [35]. While these platforms combine ideas of crowdsourcing and video annotations, they are not designed specifically to help law enforcement authorities with surveillance video analysis and alerting systems. Developing and deploying crowdsourced annotation tools for law enforcement involve many issues [36], such as protecting the video contents and allowing freedom of speech to annotators while avoiding chaos and protests are some potential issues. Therefore, the use of commercial crowdsourcing platforms such as *MTurk* for annotating surveillance videos may cause some serious problems and make the investigation at hand even tougher. Many platforms are open to public. This has led to a rising concern about the privacy of video data. Many do not differentiate volunteers based on their ability and experience. They also lack advanced administrative controls causing the number of views per video varying significantly. This may result in videos analyzed by less than adequate number of people or not analyzed at all. Most commercial tools do not use any machine learning technique with crowdsourcing model, so the analysis results provided by the crowd and that by a machine are often aggregated independently and manually. The processing resources available at the clients' side (members of the crowd) are not efficiently utilized to make annotations. The annotation process is usually tedious due to the lack of interactivity, making it less attractive to volunteers. This results into significantly higher cost of crowdsourcing-based analysis.

In this paper, we present an intelligent crowdsourcing system for forensic analysis of surveillance video that consist of recorded videos as a part of search and rescue missions or large scale investigation tasks. Our system is designed to enhance crowdsourcing by incorporating human detection, re-identification and tracking methods. At the core of our system is a hierarchal pyramid model to distinguish the ability, experience and performance of crowd members. By providing this system, forensic video analysis is made more efficient, which leads to a faster intervention by law enforcement officers or search and rescue personnel.

2. OUR PROPOSED METHOD

Our goal is to provide law enforcement authorities and search and rescue personnel with a web-based video annotation system capable of a rapid forensic analysis of surveillance video using crowdsourcing methods. This system allows the integration of machine learning techniques to assist the crowdsourcing. We previously designed a web-based video annotation system as a stand-alone crowdsourcing platform to help law enforcement [31]. In this paper, we improve it by incorporating human detection and tracking methods. We present a hierarchal pyramid model to distinguish the ability, experience and performance of the crowd members.

Figure 1 depicts our proposed approach. Our system enables "administrators" to upload a set of videos as one investigation task or a search and rescue mission. Once the videos are uploaded, the administrator can specify the type of task under investigation. This can be one of the typical offenses/events such as "assault," "battery," "missing person" or "abandoned baggage." From an administrative point of view our proposed system operates in an autonomous way to produce the final result of the crowdsourcing analysis in the form of a set of video segments specifying the events of interest as one storyline. We use videos that may contain offenses related to human activity. The system is intended to identify the person, called as a suspect committing an offense from a video and subsequently identify all other videos during which the suspect reappears.

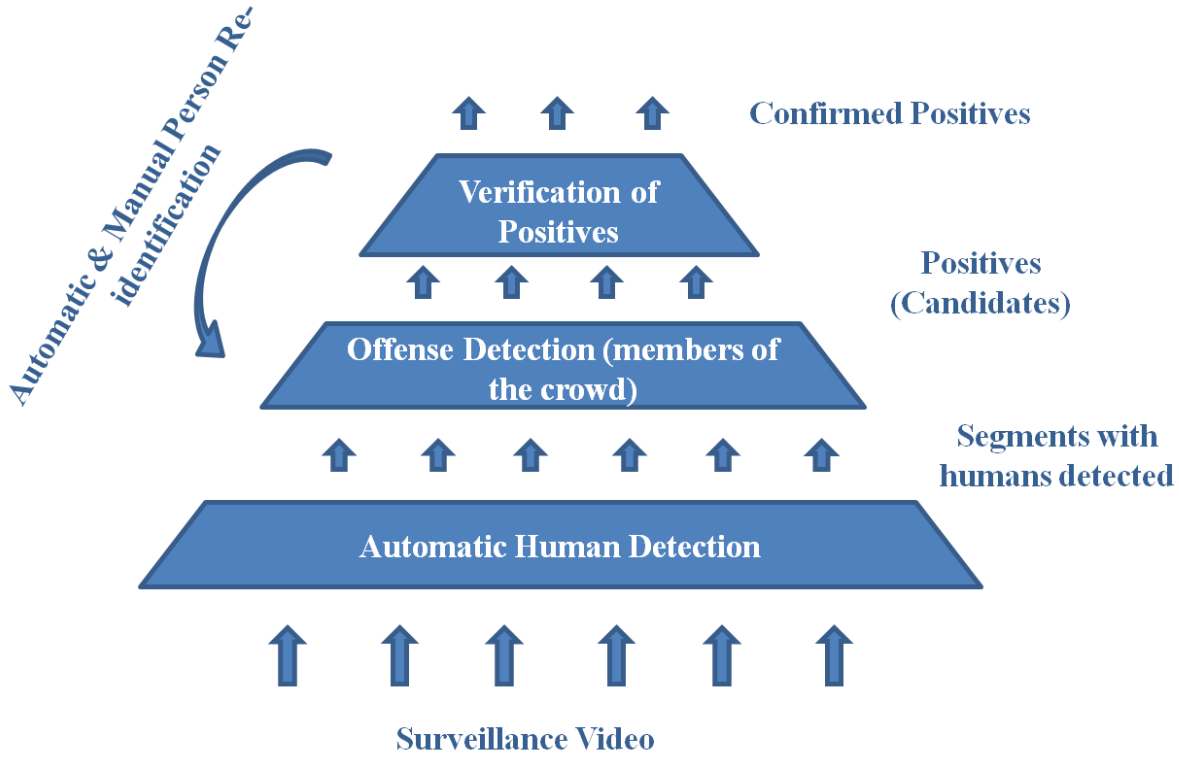


Figure 1. Our Proposed Approach.

As shown in Figure 1, our adaptive pyramid crowd model consists of several layers based on members experience and performance. Higher layers in the pyramid validate the output from lower layers. The forensic analysis starts at the base of the pyramid with a surveillance video. An automatic human detection method is used to identify videos that consist of human appearances. In the second layer of the pyramid the members of the crowd analyze video using a web-based annotation platform. They can watch for specific offenses as instructed by the law enforcement.

Our web-based annotation platform is implemented using JavaScript and HTML5 elements. This tool enables members of the crowd to watch videos and annotate them with pre-defined labels. Each annotation consists of a time interval and a bounding box capturing the suspect. To facilitate the annotation process, an object tracking

method is implemented to run on the client browser (member of the crowd) and tracks the person of interest. This helps generate multiple shots of the suspect with various poses. In cases when the tracking method fails, an optional manual tool can be used to annotate the video with shots of the suspect. The output of this process is routed to the next level of the pyramid to highly performing members of the crowd. More experienced members of the crowd are asked to validate the positives generated by the less experienced. This helps refine the output as confirmed positives are processed for person re-identification. Assignment of crowd members into pyramid layers depends on the crowd size and performance history. To keep track of performance the factors considered are: training module performance, tasks validated by higher layers, number of tasks executed and how the member results compare against the crowd average. We assume the crowd averaged results are reasonably accurate as the theory "wisdom of the crowd" [18] indicates. Using training and teaching modules, administrators can define specific video labels and upload training and teaching videos. A teaching video demonstrates the annotation process and explains what constitutes a specific offense or event. A training video allows members of the crowd to practice the annotation process.

The next step in the forensic analysis is to find all reappearances of the suspect in the surveillance video. To achieve this goal, automatic and manual analysis are performed in parallel. A task is created to look for the suspect in each video with humans detected. Less experienced members are assigned the tasks and more experienced members validate the output. Parallel to manual analysis, automatic person re-identification is started. All positives are also validated by experienced members. Confirmed offenses along with confirmed reappearances constitute the final output of the forensic analysis. This model can be also customized to specific investigation or applications. Instead of using generic human detection method, human detection for aerial images can be implemented for search and rescue missions with surveillance video being recorded from helicopters. Automatic anomaly detection, such as abandoned baggage, can also be incorporated in the model.

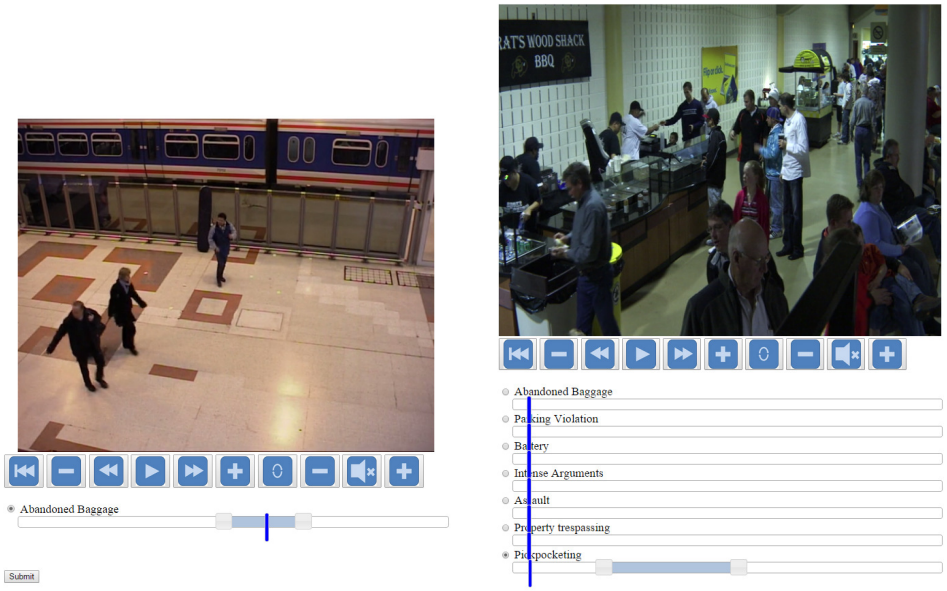
2.1 Web-Based Annotation Platform with Fast Object Tracking

It is important for the success of a crowdsourcing model that the crowd members are provided with a user-friendly annotation platform. Our tool is implemented using JavaScript and HTML5 elements that offer a user-friendly interface for the crowd. It enables members of the crowd to temporally and spatially annotate videos, using a simple click-to-add functionality. A temporal annotation is a time interval during which an offense is taking place. A spatial annotation is a snapshot that locates the suspect in a video frame. This annotation is done using adjustable rectangular boxes. Thus, the crowd members are expected to provide the temporal annotation for the specified offense and snapshots from same or different video frames showing an appearance of the suspect. These annotations are used for the subsequent automatic processing.

Figure 2(a) shows an example of the annotation interface with one slider corresponding to the offense: "Abandoned baggage." When a crowd member start a new task, a video from the collection of available surveillance video content for that investigation task is displayed with usual video player functions such as play, pause, forward, reverse and volume control. Displayed below the video player functions is a list of sliders representing specific labels corresponding to their roles. The crowd members can create a new annotation by selecting a slider with a label and clicking on the video content. This creates a highlighted time interval on the slider. Both ends of the interval can be dragged to change the duration. Figure 2(b) shows an examples of the annotation interface with several sliders corresponding to the offenses. Spatial annotations and textual comments can be added using simple click functions. The second step in the annotation process is to initialize the object tracker. Figure 3 shows an example of the annotation interface with the tracker initialized and five snapshots are automatically generated. The web-based tracker implementation is based on a method that is not computationally complex to allow real-time operation. Corners are detected according to the features from accelerated segment test (FAST) [37], while the corner point description and matching are based on binary robust independent elementary features (BRIEF) [38].

2.2 Human Detection and Re-Identification

We use an OpenCV [39] implementation of the histograms of oriented gradients (HOG) [40] combined with linear support vector machine (SVM) classifier [41]. The classifier is trained with the INRIA person dataset [42]. The goal of doing person re-identification is to find all the reappearances of a confirmed suspect in the surveillance



(a) (b)

Figure 2. Web-Based Annotation Platform.

video. It uses snapshots of the suspect to match with the humans detected in several videos. We use our implantation of person re-identification based on color and texture features. Each shot of a suspect is resized to a fixed height of 160 pixels while maintaining the aspect ratio. Then, the image is split into eight horizontal strips. For each strip, we extract four texture features and nine color features. The texture features are the following parameters of the gray level co-occurrence matrix (GLCM) [43]: "Correlation," "Homogeneity," "Energy" and "Contrast" [43]. Hue, saturation and value (HSV) color space is transformed into a one-dimensional space in a similar fashion as in [44]. Finally, the 9 values of a 9-bin histogram are used as the nine color features. The concatenation of all features from all horizontal strips form a features vector that is used for classification. Euclidean distance is used for classification. Many advanced methods have been proposed for person re-identification in which pose estimation and on-line feature selection are addressed. However, rapid analysis of surveillance video is a key requirement for forensic purposes and the computational complexity is a major factor for our choice of the re-identification techniques.

3. EXPERIMENTAL RESULTS

For our experiments we used the following publicly available surveillance video datasets: BEHAVE [45], Lunds University traffic dataset*, PETS 2006, PETS 2007 series datasets and videos from public safety communication research (PSCR) laboratory [46]. We transcoded the original videos using FFmpeg to .mp4 format.

We tested the overall performance of the system by conducting a forensic investigation. Table 1 summarizes the experiment details. The crowd consisted of 10 members varying in their abilities and performance history.

Table 1. Investigation Details

Number of surveillance videos	113
Total duration	115 minutes
Offense type	Abandoned baggage
Number of videos showing offense	1
Number of suspect reappearances	3

*Dataset available from the Video analysis in traffic research project, Lunds University, funded by The Swedish Governmental Agency for Innovation Systems (VINNOVA).

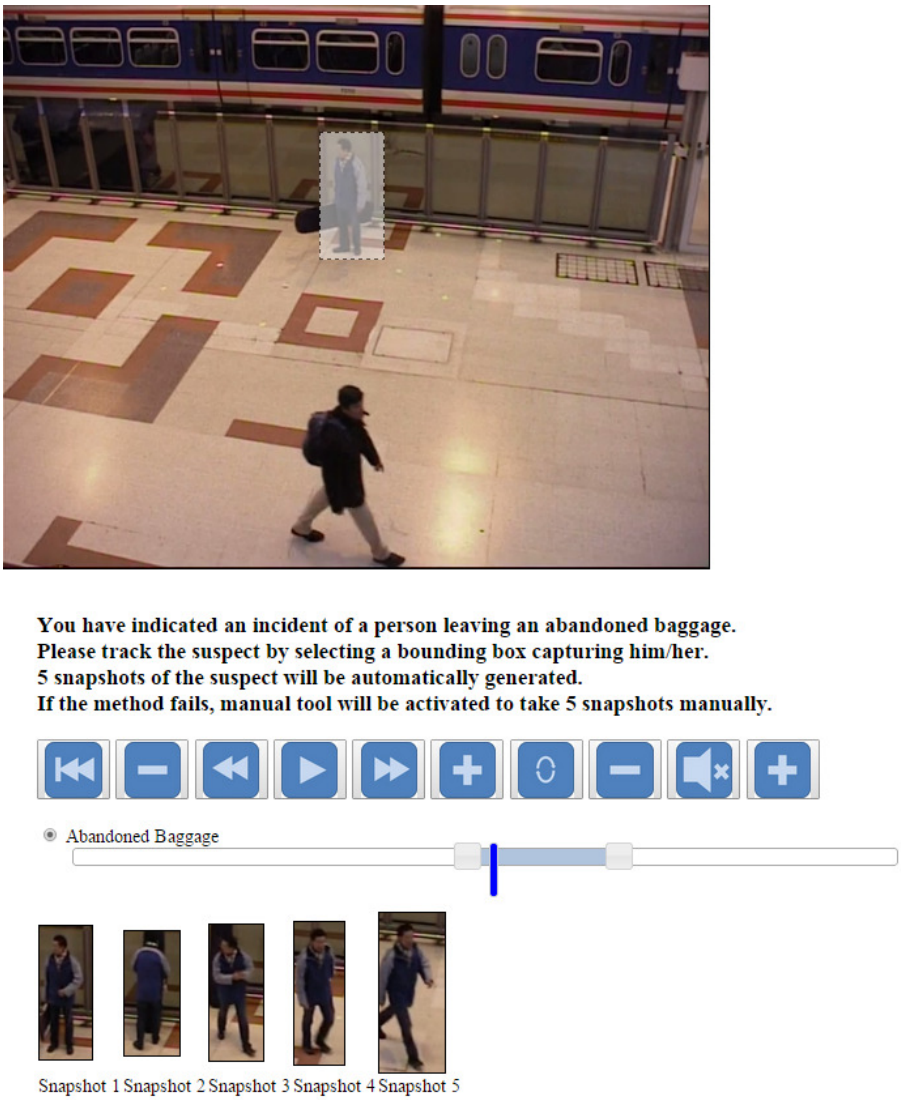


Figure 3. Web-Based Annotation Platform with Fast Object Tracking.

We believe that the surveillance video used for this experiment closely resemble a real-life scenario. The video content includes crowded scenes in stadiums, hallways, parking lots, stores and building entrances. An ideal result would be the identification of the suspect and all the three other reappearances in the surveillance video. For this experiment, the pyramid model was configured to have 8 members of the crowd in the first level of the pyramid and two experienced members in the second level. Table 2 summarizes the human detection results. Negatives are not a part of any further analysis. Only the segments with humans detected are selected

Table 2. Human Detection Results

Number of true positives	83
Number of false positives	13
Number of true negatives	15
Number of false negatives	2

for further analysis, thus saving crowd and machine resources. The presence of any false negatives is a significant concern. To overcome this, members of the first level of the pyramid can be split into two subgroups. One subgroup to investigate videos with humans detected and the other subgroup to validate the results of human detection. Another solution is to have a set of advanced human detection methods. This results in a lower probability of producing a false negative.

As reposted in Table 2, all 96 positives (83 + 13) were analyzed by the crowd. Two positives (related to abandoned baggage) were produced. One is a true positive and the second is a false positive. Following the structure of the pyramid these two positives were validated by the experienced members. The true positive was confirmed while the false positive was rejected. A confirmed positive is used for person re-identification to identify all the reappearances of the suspect. Manual and automatic re-identification tasks were created. Table 3 summarizes the human re-identification results for the automatic method.

Table 3. Human Re-Identification - Automatic

Number of true positives	1
Number of false positives	5
Number of true negatives	89
Number of false negatives	1

One out of the three reappearances were correctly identified using automatic person re-identification. Following the system structure, the results were validated by the experienced members. The true positive was confirmed while the five false positives were rejected. Table 4 summarizes the human re-identification results when performed by crowd members.

Table 4. Human Re-Identification - Using Crowd

Number of true positives	3
Number of false positives	2
Number of true negatives	91
Number of false negatives	0

Three true reappearances were identified and confirmed by the crowd. Two wrong reappearances were initially declared as positives but shortly rejected by experienced members. The final outcome of the experiment includes the correct 3 reappearances along with the main event of interest. This experiments demonstrates that it is possible to systemically incorporate machine learning methods with crowdsourcing to enhance the overall performance and speed of forensic analysis. The proposed model can be adapted to various applications and integrated with more machine learning tools.

4. CONCLUSION AND FUTURE WORK

In this paper, we proposed a crowdsourcing system for forensic analysis of surveillance video that includes the video recorded as part of search and rescue missions and large-scale investigation tasks. We used a hierarchal

pyramid model to distinguish the crowd members based on their ability, experience and performance record. Our proposed model is designed to enhance crowdsourcing by incorporating human detection, re-identification and tracking methods. The experimental evaluation demonstrates that a systematic incorporation of machine learning with crowdsourcing can improve the over all performance. In future, we plan to incorporate advanced machine learning methods with our model.

REFERENCES

[1] M. Valera and S. Velastin, "Intelligent distributed surveillance systems: a review," *IEE Proceedings of Vision, Image and Signal Processing*, vol. 152, no. 2, pp. 192–204, April 2005.

[2] N. Haering, P. Venetianer, and A. Lipton, "The evolution of video surveillance: an overview," *Machine Vision and Applications*, vol. 19, no. 5-6, pp. 279–290, September 2008.

[3] M. Shah, O. Javed, and K. Shafique, "Automated visual surveillance in realistic scenarios," *IEEE Transactions on Multimedia*, vol. 14, no. 1, pp. 30–39, January 2007.

[4] H. Dee and S. Velastin, "How close are we to solving the problem of automated visual surveillance?" *Machine Vision and Applications*, vol. 19, no. 5-6, pp. 329–343, September 2008.

[5] B. Zhani, D. Monekosso, P. Remagnino, S. Velastin, and L. Xu, "Crowd analysis: A survey," *Machine Vision and Applications*, vol. 19, no. 5-6, pp. 345–357, October 2008.

[6] J. Aggarwal and Q. Cai, "Human motion analysis: A review," *Proceedings of the IEEE Nonrigid and Articulated Motion Workshop*, pp. 90–102, June 1997, San Juan, Puerto Rico.

[7] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *ACM Journal of Computing Surveys*, vol. 43, no. 3, pp. 1–43, April 2011.

[8] P. Turaga, R. Chellappa, V. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1473–1488, September 2008.

[9] S. Srivastava and E. J. Delp, "Standoff video analysis for the detection of security anomalies in vehicles," *Proceedings of IEEE Applied Imagery Pattern Recognition*, pp. 1–8, October 2010, Washington, DC.

[10] S. Srivastava, K. K. Ng, and E. J. Delp, "Co-ordinate mapping and analysis of vehicle trajectory for anomaly detection," *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 1–6, July 2011, Barcelona, Spain.

[11] S. Srivastava and E. J. Delp, "Video-based real-time surveillance of vehicles," *Journal of Electronic Imaging*, vol. 22, no. 4, pp. 041 103–1–16, October 2013.

[12] S. Srivastava, K. K. Ng, and E. J. Delp, "Crowd flow estimation using multiple visual features for scenes with changing crowd densities," *Proceedings of the 8th International Conference on Advanced Video and Signal-Based Surveillance*, pp. 60–65, August 2011, Klagenfurt, Austria.

[13] K. Yang, E. J. Delp, and E. Du, "Categorization-based two-stage pedestrian detection system for naturalistic driving data," *Signal, Image and Video Processing*, vol. 18, no. 1, pp. 135–144, October 2014.

[14] B. Delgado, K. Tahboub, and E. J. Delp, "Automatic detection of abnormal human events of train platforms," *Proceedings of the IEEE National Aerospace & Electronics Conference*, June 2014, Dayton, OH.

[15] S. Khan and M. Shah, "Tracking multiple occluding people by localizing on multiple scene planes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 3, pp. 505–519, April 2009.

[16] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3457–3464, June 2011, Colorado Springs, CO.

[17] J. Howe, "The rise of crowdsourcing," *Wired magazine*, vol. 14, no. 6, pp. 1–4, 2006, Dorsey Press.

[18] J. Surowiecki, *The wisdom of crowds*. New York: Random House Digital, Inc., 2005.

[19] "Amazon Mechanical turk," URL: http://www.mturk.com.

[20] "Freelancer," URL: http://www.freelancer.com.

[21] "Mob4hire," URL: http://www.mob4hire.com.

[22] M. Davis, "Media streams: an iconic visual language for video annotation," *Proceedings 1993 IEEE Symposium on Visual Languages*, pp. 196–202, August 1993, Bergen, Norway.

[23] C. Vondrick, D. Ramanan, and D. Patterson, "Efficiently scaling up video annotation with crowdsourced marketplaces," *Computer Vision-ECCV 2010, Lecture Notes in Computer Science*, vol. 6314, pp. 610–623, September 2010, Springer-Verlag, Germany.

[24] A. Francois, R. Nevatia, J. Hobbs, R. Bolles, and J. Smith, "VERL: an ontology framework for representing and annotating video events," *IEEE Transactions on Multimedia*, vol. 12, no. 4, pp. 76–86, October 2005.

[25] J. Assfalg, M. Bertini, C. Colombo, and A. Bimbo, "Semantic annotation of sports videos," *IEEE Transactions on Multimedia*, vol. 9, no. 2, pp. 52–60, August 2002.

[26] S. Vijayanarasimhan and K. Grauman, "Large-scale live active learning: Training object detectors with crawled data and crowds," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1449–1456, June 2011, Providence, RI.

[27] A. Sorokin, D. Berenson, S. Srinivasa, and M. Hebert, "People helping robots helping people: Crowdsourcing for grasping novel objects," *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2117–2122, October 2010, Taipei, Taiwan.

[28] W. Willett, J. Heer, and M. Agrawala, "Strategies for crowdsourcing social data analysis," *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pp. 227–236, May 2012, Austin, TX.

[29] S. Vijayanarasimhan and K. Grauman, "Large-scale live active learning: Training object detectors with crawled data and crowds," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1449–1456, June 2011, Providence, RI.

[30] A. Sorokin, D. Berenson, S. Srinivasa, and M. Hebert, "People helping robots helping people: Crowdsourcing for grasping novel object," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2117–2122, October 2010, Taipei, Taiwan.

[31] N. J. Gadgil, K. Tahboub, D. Kirsh, and E. J. Delp, "A web-based video annotation system for crowdsourcing surveillance videos," *Proceedings of the SPIE/IS&T Electronic Imaging, Imaging and Multimedia Analytics in a Web and Mobile World*, vol. 9027, pp. 90 270A–1–12, March 2014.

[32] A. Quinn, B. Bederson, T. Yeh, and J. Lin, "Crowdflow: Integrating machine learning with mechanical turk for speed-cost-quality flexibility," *Human Computer Interaction Lab, Technical report*, May 2010, University of Maryland, College Park, MD.

[33] J. Ribera, K. Tahboub, and E. J. Delp, "Automated crowd flow estimation enhanced by crowdsourcing," *Proceedings of the IEEE National Aerospace & Electronics Conference*, June 2014, Dayton, OH.

[34] R. Brantingham and A. Hossain, "Crowded: a crowd-sourced perspective of events as they happen," *Proceedings of the IS&T/SPIE Conference on Defense, Security, and Sensing*, pp. 87 580D–1–8, February 2013, Burlingame, CA.

[35] C. Vondrick, D. Patterson, and D. Ramanan, "Efficiently scaling up crowdsourced video annotation," *International Journal of Computer Vision*, vol. 101, no. 1, pp. 184–204, January 2013.

[36] D. Brabham, *Crowdsourcing*. Cambridge, Massachusetts: The MIT Press, 2013.

[37] R. E, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, November 2010.

[38] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," *Proceedings of the European Conference on Computer Vision*, pp. 778–792, September 2010, Crete, Greece.

[39] "OpenCV," URL: http://www.opencv.org.

[40] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 886–893, June 2005, San Diego, CA.

[41] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, September 1995.

[42] "INRIA Person Dataset," URL: http://pascal.inrialpes.fr/data/human/.

[43] R. M. Haralick, K. Shanmugam, and H. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man and Cybernetics*, no. 6, pp. 610–621, November 1973.

[44] J. Ma, "Content-based image retrieval with hsv color space and texture features," *International Conference on Web Information Systems and Mining*, pp. 61–63, November 2009, Shanghai, China.

[45] University of Edinburgh, The BEHAVE Dataset, http://groups.inf.ed.ac.uk/vision/.
[46] "Public Safety Communication Research," URL: http://www.pscr.gov.

AUTOMATED CROWD FLOW ESTIMATION ENHANCED BY CROWDSOURCING

Javier Ribera[⋆] *Khalid Tahboub*[†] *Edward J. Delp*[†]

[⋆] Universitat Politècnica de Catalunya, Barcelona, Spain
[†] Video and Image Processing Lab (VIPER), Purdue University, West Lafayette, Indiana USA

ABSTRACT

Video surveillance systems that contain a large number of cameras makes the continuous monitoring of the video feeds nearly an impossible task. If the information from these many cameras is to be exploited automatic video analytic techniques must be developed. In this paper, we present improvements to a previously developed crowd flow estimation method. We use crowdsourcing techniques to enhance the performance. An experimental evaluation is conducted using publicly available datasets.

Index Terms— crowd flow estimation, crowd analysis, people counting, crowdsourcing

1. INTRODUCTION

Video surveillance systems are widely deployed with the number of cameras increases exponentially. The increasing number of cameras makes the continuous human monitoring of video nearly impossible. Many video analytic techniques have been developed to provide automatic analysis of the video data [1, 2].

One type of analysis is measuring attributes of "crowds." These include the number of people in a crowd, the direction that a crowd is moving, and abnormal events occurring in the crowd (e.g. fighting). In this paper we examine another crowd attribute known as crowd flow. Crowd flow is the process of estimating the number of people crossing a specific spatial zone. One measure of crowd flow is crowd density which is the number of people moving through a region in a given period of time [3].

There are various methods for crowd flow estimation [3]. Direct methods involve tracking all individuals and estimating the number of people crossing a specific region. Scalability becomes an issue when dealing with large crowds. Indirect methods estimate the crowd density using image features. In [11] a linear relationship between the number of pedestrians and the number of pixels in the foreground segmentation is assumed based on a constant level of occlusion. In [12, 13] the level of occlusion is related to the texture of the image. In [14] we used this assumption and incorporated texture features to consider changing crowd densities. In [15] the use of features such as edge orientation and blob size histograms is described. In [16] the number of people in the crowd is estimated using geometric, edge, and texture features.

The use of video analytics face significant challenges such as object occlusions, shadows, sudden and gradual scene illumination changes. Automatic methods are not perfect and maintaining an acceptable level of accuracy is often a difficult task. Crowdsourcing may be an effective solution to problems that involve cognitive tasks. By crowdsourcing here we mean the use of a group of people (e.g. expert observers) to observe the surveillance video and "help" the automatic analysis method perform better.

Crowdsourcing was defined by J. Howe in 2006 [4]. Since then, there has been many studies utilizing human intelligence to help machines perform better. Using crowds are often implemented through commercial platforms such as the Amazon Mechanical Turk (MTurk) [5], Freelancer [6], and Mob4hire [7]. In [8], object classifiers are trained and continuously refined by reaching out to members of the crowd through MTurk. In [9], a system for enhancing machine learning using crowdsourcing is proposed. In [10], we described a web-based system for the annotation of surveillance video in an organized and controlled way.

In this paper, we describe an approach to enhance the performance of crowd flow by utilizing crowdsourcing. To avoid confusion, in this paper we shall refer to the group of people observing/watching the surveillance video (e.g. expert observers) and helping with the crowd flow analysis as the "observation crowd" or "o-crowd." The crowd of people being observed by the cameras from which we want to estimate crowd flow shall be known as the "crowd." Our proposed method "asks" or uses the o-crowd in cases where the automatic crowd flow is uncertain in making a particular decision. In this paper, we improve the performance of our previously developed approach described in [14] with enhancements to the automatic detector and incorporation of crowdsourcing.

2. CROWD FLOW ESTIMATION

Our goal is to estimate the number of people crossing a desired region of the image in a given time interval, i.e. crowd flow, and we focus on pedestrians. We use an approach we developed earlier in which two regions are defined [14]: a tripwire which is a region through which people passing are counted and a larger Region Of Interest (ROI) surrounding the tripwire. The idea is to assume that the number of people crossing the tripwire is proportional to the foreground pixels in the region.

$$v_N = \frac{\tilde{S}_N}{C} \tag{1}$$

where v_N is the number of people that have crossed the tripwire during a stable period of N frames. A stable period is defined as a set of consecutive frames with the same level of crowdedness. Crowdedness is related to occlusions of persons and the crowd density level, that is the number of persons present in the scene. As long as the crowdedness level stays the same, we assume that the number of foreground pixels is linearly proportional to the number of people crossing the region. $\tilde{S}_N$ is the weighted accumulation of foreground pixels related to moving objects inside the tripwire and during this period. Moving object detection is done using the background sub-

This work was partially supported by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0001.
Address all correspondence to Edward J. Delp, ace@ecn.purdue.edu

traction method proposed in [17]. C is the scaling constant associated with this level of crowdedness and represents the number of foreground pixels seen per person at a given level of crowdedness. Typically, high crowd density level has more occlusions with people. As a result, the number of foreground pixels seen per person is less compared to the case of low crowd density level. Therefore, C is dependent on the crowdedness level.

Moving object detection is only done inside the tripwire region. The background model is initialized during the first 30 frames during which the scene is empty. The result of the background subtraction is a foreground mask $I_n(x, y)$ for every frame n of the video feed. $I_n(x, y)$ is 1 for pixels belonging to moving objects and 0 otherwise. The foreground pixel accumulation S_N is determined by:

$$S_N = \sum_{n=1}^{N} \sum_{x,y \epsilon \Re} I_n(x, y) \tag{2}$$

where $\Re$ denotes the tripwire region. Due to perspective distortions, the blob size of an object in the foreground mask varies according to the distance from the camera. To account for this, a weighting scheme is used. The weighted accumulation of foreground pixels is determined by:

$$\tilde{S}_T = \sum_{n=1}^{N} \sum_{x,y \epsilon \Re} I_n(x, y) \cdot \omega(x, y) \tag{3}$$

where $\omega(x, y)$ is the weighting function. The weighting function only needs to be defined for pixels in the tripwire region. Our proposed weighting function is: First, the system operator/user is asked to draw a quadrilateral which corresponds to a rectangle in the real world. The quadrilateral is defined by its four sides L_1, L_2, L_c and L_f as shown in Figure 1. The operator is also asked to indicate the closer and distance sides (L_c and L_f respectively). Due to perspective, L_1 and L_2 appear to intersect at the vanishing point F. Second, for each point (x, y) inside $\Re$, we find the point Q on L_c which corresponds to the intersection of L_c and the line passing through (x, y) and F. Third, we define a segment S_c centered at Q and laying on L_c, S_c that has a fixed predefined length W_c. Fourth, the segment S_c is projected towards the vanishing point F such that a segment S_f is found to have the point (x, y). The segment S_f is parallel to S_c with length W_f. Due to perspective, W_f is not equal to W_c, even though they have the same length in real world. The weighting function at (x, y) is then defined as:

$$\omega(x, y) = \frac{W_c}{W_f} \tag{4}$$

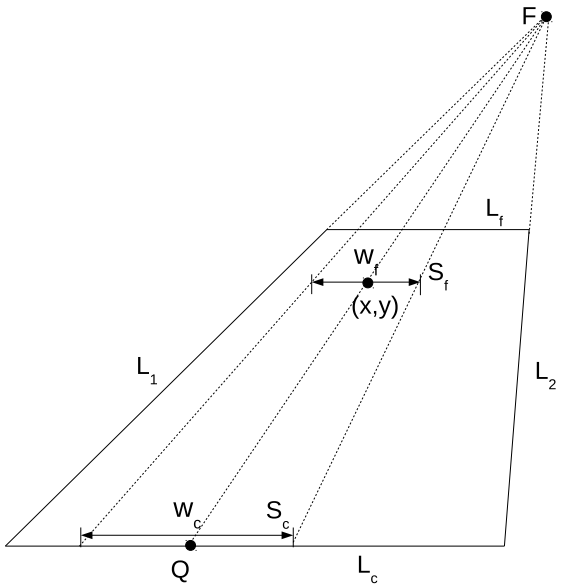


Fig. 1. Illustration of the Weighting Scheme.

3. ESTIMATION OF CROWDEDNESS

Large crowd density results in a large number of occlusions which means a reduced number of foreground pixels per person. Therefore, the level of crowdedness should be estimated in order to scale the foreground pixel count properly. In order to estimate the crowd density, we use the approach described in [12] in which texture features and crowd density are shown to be related. In [12] a crowded scene presents a fine texture, while a sparse scene has a coarse texture. We use a 16-D texture feature based on the Gray Level Co-occurrence Matrix (GLCM) [18].

GLCM models a texture by characterizing the probability that a pixel with a specific gray level is adjacent to another specific gray level. The GLCM can be thought of as an estimate of the 2D joint probability distribution function of the pixels in the image. Using the ROI previously defined, we create four GLCM matrices. Each matrix represents a different direction for the adjacent pixels: right (0^o), top-right (45^o), top (90^o), and top-left (135^o). Then for each matrix we extract four scalar features: energy (5), entropy (6), homogeneity (7) and contrast (8).

$$Energy(P) = \sqrt{\sum_{i,j} p_{ij}^2} \tag{5}$$

$$Entropy(P) = -\sum_{i,j} p_{ij} \log p_{ij} \tag{6}$$

$$Homogeneity(P) = \sum_{i,j} \frac{p_{ij}}{1 + (i - j)^2} \tag{7}$$

$$Contrast(P) = \sum_{i,j} p_{ij} (i - j)^2 \tag{8}$$

"Energy" is defined to be the square root of the sum of squared elements in GLCM and is maximum for a constant image. Contrast is a measure of the intensity contrast between a pixel and its neighbors over the whole image. Contrast is minimum for a constant image. Homogeneity measures the closeness of the distribution of elements in the GLCM to the diagonal of the matrix and is 1 for a diagonal GLCM. Entropy is a statistical measure of randomness. We use these 16 scalar features to construct a 16-D feature vector, t_n, that represents the texture of one frame n.

Let L be the number of crowdedness levels specified by the operator for a video. We assume that each crowdedness level has a unique texture. The training stage, as described later, aims to find L texture feature vectors which represent the L levels of crowdedness. During the training stage, the o-crowd (the expert observers) is asked to classify many frames (textures) into the L levels of crowdedness. The first output of the training stage is L texture feature vectors (τ_1,

τ_2, ..., τ_L) representing L levels of crowdedness. Each vector is used as a reference vector for the corresponding level of crowdedness. From now on we shall call these L texture feature vectors as reference vectors. As mentioned earlier, the scaling factor C relating the number of foreground pixels to the number of people crossing the tripwire region is dependent on the level of crowdedness. During the training stage, the o-crowd is asked to count the number of people crossing the tripwire region during a short period of time and C_i is determined accordingly. C_i, $i\epsilon\{1, 2, ..., L\}$, is the second output of the training stage. When the method is being used to count people, a texture feature vector representing the texture of the ROI, t_n, is determined for each frame. The first step is to find the level of crowdedness. This is a classification problem in which we classify a 16-D texture feature vector into one of the levels of crowdedness represented by the L reference vectors. The closest reference vector is used for classification, this is a K-Nearest Neighbors (KNN) classifier in which $K = 1$. When the level of crowdedness is determined, the second step is to use the associated scaling factor C_n for this level of crowdedness and estimate the number of people as determined in Equation (9), which is a combination of Equations (1) and (3):

$$\upsilon_N = \sum_{n=1}^{N} \frac{\sum_{x,y\epsilon\Re} I_n(x,y) \cdot \omega(x,y)}{C_n} \tag{9}$$

4. ENHANCEMENT USING CROWDSOURCING

The classification step and the resulting scaling factor of the current ROI is critical for performance. To enhance the classification performance, we use crowdsourcing to reduce uncertainty. For a frame n, the distance between the texture feature vector t_n and the nearest reference vector, d_1, might be very comparable to the distance between t_n and the second nearest neighbor, d_2. In this case, the classifier will choose the level of crowdedness corresponding to the nearest reference vector even if the difference between the two distances $(d_2 - d_1)$ is very small. To overcome such uncertain classification decisions, we define "uncertainty" μ as the ratio $\frac{d_1}{d_2}$ to represent the uncertainty of the classification decision. By definition $d_1 \leq d_2$ and therefore $\mu\epsilon[0, 1]$. Larger values of μ represent greater uncertainties as d_1 and d_2 are comparable. Therefore, the o-crowd is asked to classify uncertain frames whenever "uncertainty" (represented by μ) is greater than a predefined threshold, α. This is then formally done when Equation (10) is fulfilled.

$$\mu = \frac{d_1}{d_2} > \alpha \tag{10}$$

α represents the maximum uncertainty allowed and should be between 0 and 1. For example, $\alpha = 0.5$ requires the feedback of the o-crowd whenever the distance between t_n and the second closest reference is less than twice the distance to the closest reference vector. We call α the "crowdsourcing parameter." It is related to how often we refer to the o-crowd (i.e. ask the o-crowd for help). This defines a "certainty area" around a training vector. Feature vectors outside all certainty areas are automatically referred to the o-crowd. The video frame corresponding to this t_n is shown to the o-crowd and they are asked to estimate its level of crowdedness and the number of people in the crowd. Figure 2 illustrates the "certainty area" around two reference vectors.

In Figure 2 the two reference vectors are represented by the cross signs. 2-D representations of the 16-D feature vectors are used for illustration purposes. Each circle represents a certainty area around the reference vectors for a specific value of α. When $\alpha = 1$, the entire feature space is divided into two regions, each region is a certainty area around one of the two reference feature vectors. This corresponds to the case when crowdsourcing is not used. Choosing $\alpha = 0.8$ reshapes the certainty areas to circles, each circle surrounding one of the reference feature vectors. During testing, any video frame which has a resulting texture feature vector outside both circles will be referred to the o-crowd for evaluation. Therefore, by controlling the size of each circle we control how often we ask the o-crowd for help. Choosing $\alpha = 0.6$ or 0.4 clarifies this case as the size of the certainty areas become smaller and it is likely that the o-crowd is engaged more often.

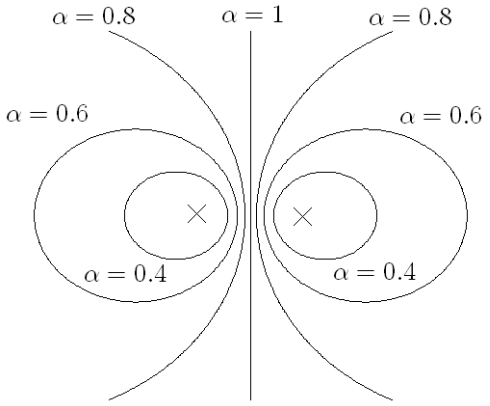


Fig. 2. Certainty areas around two reference vectors corresponding to four values of α.

When using crowdsourcing we make sure not to ask the o-crowd for redundant information. This is particularly important because consecutive video frames are likely to have similar feature vectors. Therefore, we store the t's for which the o-crowd has provided feedback and define areas centered at the t's in which future feature vectors in next frames are not referred to the o-crowd. Consider a feature vector t_{test} that falls outside the certainty areas. Then, the o-crowd classifies its level of crowdedness and provide the number of people. t_{test} is stored as a new reference vector and its scaling factor is determined. Future classifications will perform better by taking into consideration this new reference vector. However, the initial training process relies on more samples and is more trustworthy. To take this into account, certainty areas surrounding o-crowd reference vectors are scaled down. This is done by reducing α by a factor of 0.9 for reference vectors produced by the o-crowd.

5. EXPERIMENTAL RESULTS

5.1. Training

As discussed earlier, the number of foreground pixels per person is dependent on the level of crowdedness. Therefore, the training process aims to train the classifier to classify the 16-D texture feature vectors into one of the levels of crowdedness. The first output of training is L reference vectors (texture feature vectors) (τ_1, τ_2, ..., τ_L) representing L levels of crowdedness. Each vector is used as a reference vector for the corresponding level of crowdedness. The second output is the scaling factor C_i for each level i of crowdedness, $i\epsilon\{1, 2, ..., L\}$. The dataset used for testing is the UCSD

pedestrian dataset [19]. For our experiments the o-crowd consists of two members of our laboratory.

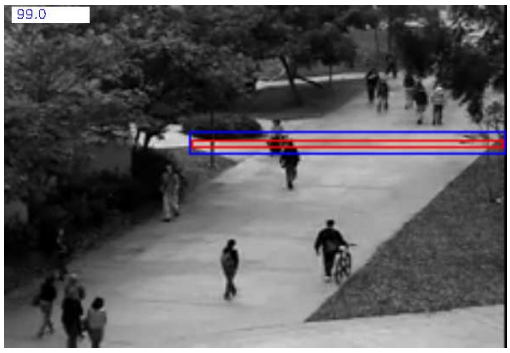


Fig. 3. A video frame from our test dataset [19] including the Tripwire region $\Re$ and the ROI surrounding it.

First, the system operator is asked to mark the Tripwire region $\Re$ and the ROI surrounding it. The perspective weighting function is determined accordingly. Figure 3 shows an example from our test dataset [19] including the Tripwire region $\Re$ and the ROI surrounding it. Next, 40 video frames are chosen randomly from the training video segment, which is the first 3 minutes of the dataset video. For each frame, An o-crowd member is asked to classify each video frame into one of the L levels of crowdedness. τ_l's are estimated as the average of the training vectors for each corresponding level. To find the scaling factor C_i for each level, we find the longest stable period for each level of crowdedness. A stable period is a set of consecutive frames with the same level of crowdedness. L video segments corresponding to each level l are shown during training, an o-crowd member has to count the number of persons crossing the tripwire region. The scaling factor C_l for each level of crowdedness is determined by Equation (1).

5.2. Testing

The dataset used for testing is the UCSD pedestrian dataset [19]. This dataset is suitable for testing purposes since it is long enough and contains different crowd density levels. We used 8 minutes of video and segmented it into 6 clips, the first clip is used solely for training and is 3 minutes long. The remaining 5 minutes are segmented into 5 clips each of 1 minute duration. The selected number of crowdedness levels is $L = 3$. In [14], it was noted that the larger levels of crowdedness gives better results. However, in this dataset, 3 distinct levels of crowdedness could be observed. During training and testing, moving object detection is done using the background subtraction method proposed in [17]. Learning rate represents how fast the background model is updated [20] and was set to 0.01 during both stages.

Several tests were conducted to examine various aspects of our proposed method. The first is to evaluate the performance of the automatic method with no use of crowdsourcing. The second aims to introduces contrast changes and compression attacks to the video and evaluate the performance. The third considers crowdsourcing as a solution to quality degradation. We compare the performance and evaluate the utilization of the crowd for two values of α.

5.3. Results and Discussion

The dataset used for testing does not provide ground truth information, therefore, we used our best judgment to provide the ground truth data. However, the numbers might be "± 0.5" person from the true value due to the fact that some people are crossing the tripwire at the beginning or the end of each clip. Figure 4 displays crowd flow estimation error rates for the 5 clips. Crowd flow estimation error rate is the difference between the estimated value and ground truth value as a percentage of the ground truth value. The x-axis are the clip numbers, each clip is 1 minute of duration. The average error rate for the original video quality is 9.25%. Error rates for degraded video quality are 35% higher for 3 out of 5 segments, the average error rate is 34.38%. Video quality degradation due to compression and contrast changes is of practical importance. Many surveillance cameras are connected to processing centers through wireless networks, modern video compression techniques adapt to the available resources and the video quality varies accordingly. For this experiment, H.264 compression was implemented using FFmpeg Constant Rate Factor (CRF) [21], CRF was set to 33. Contrast was boosted to 1.68 using FFmpeg eq2 filter [21]. We conclude that crowd flow estimation using texture features and foreground pixels accumulation is not robust against contrast changes and video compression.

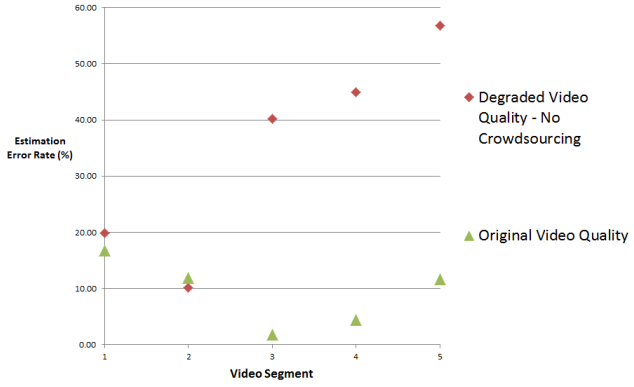


Fig. 4. Crowd flow estimation error rates for original and degraded video quality.

Figure 5 displays error rates for crowd estimation when enhanced by crowdsourcing. $\alpha = 0.6$ and 0.8 are used and the resulting error rates are significantly less than without crowdsourcing. The average error rates are 18.02% and 14.34% respectively. This demonstrates that our proposed method manages to identify uncertainties and use the crowd in an effective way to lower the error rates.

Figure 6 displays the number of crowdsourcing tasks per video segment. A crowdsourcing task is an instance in which the "uncertainty" as defined in section 4 is higher than α. This is when Equation (10) is satisfied and the o-crowd is asked to classify an uncertain frame and provide the people count. Ideally, we would like the method to learn from the o-crowd input and the number of times the o-crowd is asked for help to decrease as we progress in time. Figure 6 depicts that our proposed method achieves this and not only reduces the error rates but also trains the classifier in an effective way, such that the number of tasks decreases as we progress in time. This figure also shows that for lower value of α the o-crowd

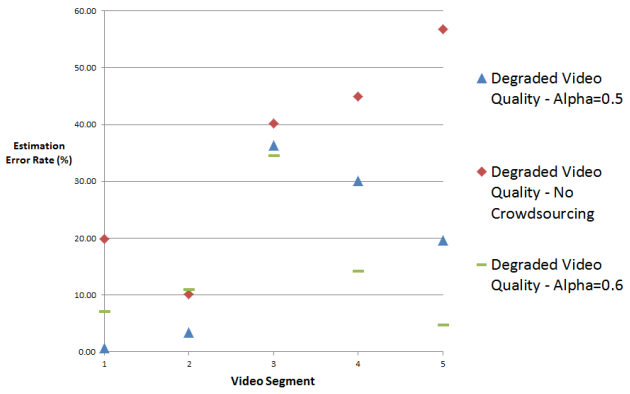


Fig. 5. Crowd flow estimation error rates for degraded video quality with and without crowdsourcing enhancement.

is engaged more often.

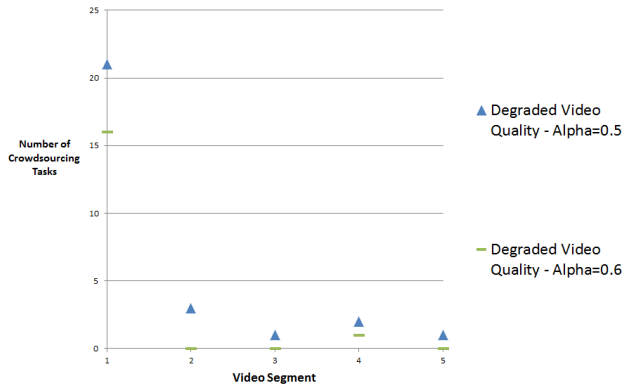


Fig. 6. Crowdsourcing tasks per video segment.

6. CONCLUSION

In this paper we presented enhancements to our previously developed crowd flow estimation method. Experimental evaluation demonstrates that our proposed method identifies uncertainties and uses crowdsourcing in an effective way. Crowd flow estimation error rate is significantly reduced and the rate at which the o-crowd is engaged decreases in time. Future extensions of our work include building a pyramid model to differentiate the accuracy of various crowd members and incorporating it in our method and the use of larger size for the o-crowd.

7. REFERENCES

[1] M. Valera and S. Velastin, "Intelligent distributed surveillance systems: a review," *IEE Proceedings of Vision, Image and Signal Processing*, vol. 152, no. 2, pp. 192–204, April 2005. doi: [10.1049/ip-vis:20041147]

[2] N. Haering, P. Venetianer, and A. Lipton, "The evolution of video surveillance: an overview," *Machine Vision and Applications*, vol. 19, no. 5-6, pp. 279–290, June 2008. doi: [10.1007/s00138-008-0152-0]

[3] B. Zhan, D. Monekosso, S. V. P. Remagnino, and L. Xu, "Crowd analysis: A survey," *Machine Vision and Applications*, vol. 19, no. 5-6, pp. 345–357, October 2008. doi: [10.1007/s00138-008-0132-4]

[4] J. Howe, "The rise of crowdsourcing," *Wired magazine*, vol. 14, no. 6, pp. 1–4, 2006, Dorsey Press.

[5] "Amazon Mechanical turk," URL: http://www.mturk.com.

[6] "Freelancer," URL: http://www.freelancer.com.

[7] "Mob4hire," URL: http://www.mob4hire.com.

[8] S. Vijayanarasimhan and K. Grauman, "Large-scale live active learning: Training object detectors with crawled data and crowds," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1449–1456, June 2011. doi: [10.1109/CVPR.2011.5995430] Providence, RI.

[9] A. Sorokin, D. Berenson, S. Srinivasa, and M. Hebert, "People helping robots helping people: Crowdsourcing for grasping novel object," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2117–2122, October 2010. doi: [10.1109/IROS.2010.5650464] Taipei, Taiwan.

[10] N. J. Gadgil, K. Tahboub, D. Kirsh, and E. J. Delp, "A web-based video annotation system for crowdsourcing surveillance videos," *Proceedings of the SPIE/IS&T Electronic Imaging, Imaging and Multimedia Analytics in a Web and Mobile World*, vol. 9027, pp. 90 270A–1–12, March 2014. doi: [10.1117/12.2042440]

[11] J. H. Yin, S. A. Velastin, and A. C. Davies, "Image processing techniques for crowd density estimation using a reference image," *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pp. 489–498, December 1995. doi: [10.1007/3-540-60793-5_102] Singapore.

[12] A. N. Marana, S. A. Velastin, L. F. Costa, and R. A. Lotufo, "Automatic estimation of crowd density using texture," *Safety Science*, vol. 28, no. 3, pp. 165–175, April 1998. doi: [10.1016/S0925-7535(97)00081-7]

[13] A. N. Marana and S. A. Velastin and L. F. Costa and R. A. Lotufo, "Estimation of crowd density using image processing," *Proceedings of the IEE Colloquium on Image Processing for Security Applications*, pp. 11/1–11/8, March 1997. doi: [10.1049/ic:19970387] London, United Kingdom.

[14] S. Srivastava, K. K. Ng, and E. J. Delp, "Crowd flow estimation using multiple visual features for scenes with changing crowd densities," *Proceedings of the 8th IEEE International Conference on Advanced Video and Signal-Based Surveillance*, pp. 60–65, August-September 2011. doi: [10.1109/AVSS.2011.6027295] Klagenfurt, Austria.

[15] D. Kong, D. Gray, and H. Tao, "Counting pedestrians in crowds using viewpoint invariant training," *Proceedings of the British Machine Vision Conference*, September 2005, Oxford, UK.

[16] A. B. Chan, Z. J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–7, June 2008. doi: [10.1109/CVPR.2008.4587569] Anchorage, AK.

[17] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern recognition letters*, vol. 27, no. 7, pp. 773–780, May 2006. doi: [10.1016/j.patrec.2005.11.005]

[18] R. M. Haralick, "Statistical and structural approaches to texture," *Proceedings of the IEEE*, vol. 67, no. 5, pp. 786–804, May 1979. doi: [10.1109/PROC.1979.11328]

[19] A. B. Chan and N. Vasconcelos, "Modeling, clustering, and segmenting video with mixtures of dynamic textures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 5, pp. 909–926, May 2008. doi: [10.1109/TPAMI.2007.70738]

[20] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, June 1999. doi: [10.1109/CVPR.1999.784637] Fort Collins, CO.

[21] "FFmpeg," URL: http://www.ffmpeg.org.

AUTOMATIC DETECTION OF ABNORMAL HUMAN EVENTS ON TRAIN PLATFORMS

Blanca Delgado[⋆] *Khalid Tahboub*[†] *Edward J. Delp*[†]

[⋆] Universitat Politècnica de Catalunya, Barcelona, Spain
[†] Video and Image Processing Lab (VIPER), Purdue University, West Lafayette, Indiana USA

ABSTRACT

Video surveillance systems that contain a large number of cameras makes the continuous monitoring of the video feeds nearly an impossible task. A transit or transportation authority usually deploys a video surveillance system to monitor and identify events in the system such as crowd behavior and crime. In this paper we present a method for automatically detecting people jumping or falling off a train platform. An experimental evaluation is described using a dataset that was recorded at a train station.

***Index Terms*—** video surveillance, automatic detection, train platform, background subtraction

1. INTRODUCTION

Video surveillance systems are widely deployed with the number of cameras increases exponentially. The increasing number of cameras makes the continuous human monitoring of video nearly impossible. Many video analytic techniques have been developed to provide automatic analysis of the video data [1, 2]. A typical surveillance system deployed by a transit or transportation authority is used to monitor and identify events in the system such as crowd behavior and crime [2]. The task of human monitoring becomes more tedious when events of interest are not probable, such as anomalous behaviors, fights, crimes and threats. Therefore, surveillance video data are usually archived for forensic purposes. Video analytic techniques have been proposed to automatically identify criminal activities and anomalous behaviors. Anomaly detection in crowded scenes, fights and abandoned baggage detection are some examples where video analytic techniques can save lives and exploit surveillance video in a proactive way [3, 4, 5, 6, 7, 8].

An excellent survey on human behavior recognition methods with respect to transit systems is presented in [9]. Core concepts including motion detection, object classification and tracking are surveyed. Human-behavior recognition is divided into single or no person interaction, multiple-person interactions, person-vehicle interactions and person-facility interactions. In [10], a motion-based image processing system for detecting dangerous situations in underground railway stations is presented. Situations of interest include overcrowding, unusual or forbidden directions of motion and stationary individual or objects.

Many cities witnessed fatal incidents where people were hit and killed by trains. In some cases the victim had suicidal motives and jumped off the train platform and in other cases were the result of criminal activities where people were pushed off the platform. To achieve our goal of detecting people moving into the track bed area we have to overcome several challenges. Some have to do with the dynamic environmental conditions in outdoor surveillance video, such as sudden or gradual illumination changes, shadows and motion resulting from uninteresting objects such as waving leaves. Other challenges result from the nature of our application where moving trains might be misclassified as moving people.

In 2005 the IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS) had a "challenge" session focused on real-time event detection. The public transportation network in Paris (RATP) [11] was asked to describe its surveillance needs and from that the Challenge of Real-time Event Detection Solutions (CREDS) was formulated to investigate events in indoor scenes, including track bed intrusion and objects on rails [12, 13, 14, 15, 16, 17].

Various methods were proposed to address the AVSS/CREDS challenges. In [12], a non-linear background update is used with region growing segmentation. From the generated blobs, features were extracted and a tracking method is used. In [13], a block based motion estimation approach is used for background estimation. A monotonic relationship between the number of observed foreground regions and crowding in order to detect overcrowding is described. This strategy was extended to identify the presence of the train. A block based motion estimation approach is also used in [16]. In order to classify different objects and events, information such as size, location, direction, shape and color are used.

3D calibration of multiple cameras is used in [14] to generate perspective information. To detect targets in the scene, a background mask is obtained, followed by region growing based segmentation of moving objects and model-based tracking. In [15], low level analysis based on features and tracking is followed by high level analysis based on principles of artificial intelligence. The low level analysis uses techniques such as motion detection, background differencing and histogram analysis.

A Gaussian Mixture Model (GMM) for background subtraction is proposed in [17] for illumination changes in an indoor subway station. To distinguish between humans and trains, shape information of blobs is used. In [18], statistical analysis of the image background are proposed and followed by region-growing in order to detect different areas. In order to detect train presence, a DC-notch filter is implemented. In [19], geographical distributions of areas are used to differentiate between different targets. To evaluate the results for the CREDS challenge, the criteria in [20] was followed. Seven warning events were defined. Each event is defined in an image zone and the results were compared against ground truth information. The characterization of tracking performance is also proposed in [21]. The RATP dataset used for the CREDS challenge along with surveillance video from various subway stations in Seoul, Korea were used in [18]. The surveillance video used in [19] were recorded in the subway in Berlin, Germany.

This work was partially supported by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0001.
Address all correspondence to Edward J. Delp, ace@ecn.purdue.edu

In this paper, we present a method to automatically detect persons moving into the "track bed" region and detect the train presence based on its motion. We use motion information in a novel way to detect trains and distinguish arriving trains from departing trains. Constrained to the track bed region, we find the most prominent corners of each frame and track them between frames to estimate speed. We take advantage of our prior information that trains move along the edge of the track bed to detect them by estimating the direction of moving objects. Our proposed method accounts for dynamic environment changes present in outdoor scenes using a GMM for the background model. The implementation is not hardware specific and can be run in real-time. An experimental evaluation is conducted with a dataset that was recorded at a local train station.

2. PROPOSED METHOD

To achieve our goal of detecting people moving into the track bed region, we detect moving objects and analyze their location in relative to the edge of the track bed. Train presence is investigated by analyzing motion vectors in the track bed region. Figure 1 shows a block diagram of our proposed method.

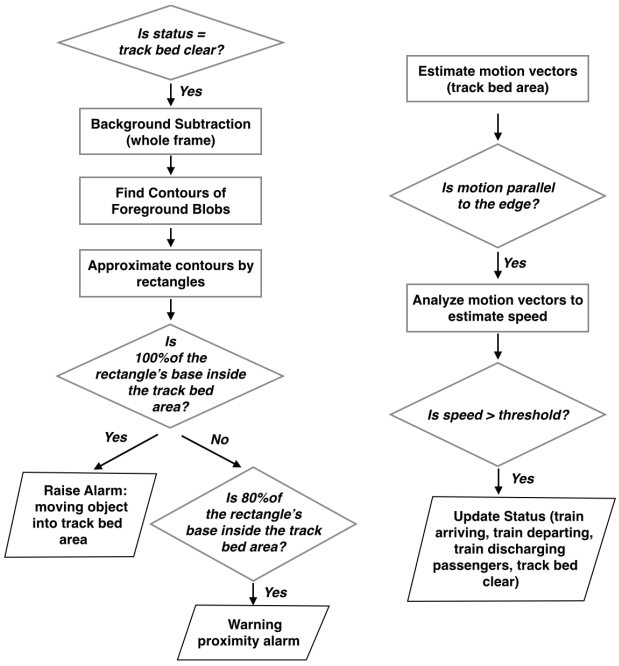


Fig. 1. Block Diagram of the Proposed Method.

It is assumed that our method "knows" where the track bed is located. A line separating the track bed and the platform is determined manually. This is done once and is the only manual interaction for our technique. Figure 2 shows an image from our test dataset. The edge of the track bed is marked manually in red.

Fig. 2. Train Platform and Track Bed Region.

2.1. Background Subtraction

We estimate the "constant" background region and then use change detection to detect people in the track bed. We use the implementation available in OpenCV [22] which is based on [23, 24]. Change detection becomes the problem of background estimation. Every pixel of the background is modeled as a random variable. The mean or average represents the actual background and its variation with respect to the mean is due to the noise introduced by the camera sensor and other sources. This noise is modeled as a Gaussian process. To account for variations introduced by dynamic environmental conditions, a Gaussian Mixture Model [25] is used as a statistical model. In [23, 24], each pixel is modeled by a mixture of K Gaussians (typically, $5 \geq K \geq 3$). Each of the K Gaussians has three dimensions to model red, blue and green (RBG) color channels. The weight of each Gaussian represents the time proportion the associated color is expected to be in the scene. Not all of the K Gaussians necessarily represent the background model. Moving objects belonging to the foreground might be incorporated in the GMM with a corresponding Gaussian. However, we assume that background pixels are more probable than foreground pixels and the weights of the background Gaussians to be greater. To determine which Gaussians belong to the background and which belong to the foreground, we weight all Gaussians in descending order and take the first B Gaussians as a model of the background of the scene, where B is:

$$B = \arg\min_{b}(\sum_{j=1}^{b} w_j > T)$$

Where w's are the weights of the Gaussian components. B is governed by the parameter T above, where $T = 0.9$ for our experiments.

The rate at which the model parameters is updated is governed by α, the learning rate [24]. For our experiments $\alpha = 0.0001$. Every new pixel is checked against the existing model components. If there is a match, the associated component is updated. If not, a new Gaussian component is created. Typically, the foreground mask consists of people and trains. The next task is to distinguish between people and trains. Morphological operations are used to remove noise and unify the blobs of the foreground mask. Opening is used for removing small noisy points, while closing is used to merge the blobs of the foreground [26]. An elliptical structuring element is used for both operations.

2.2. Train Presence

Motion information can adequately be used to detect the train presence. Train motion is limited to the track bed area and is parallel to

the platform edge. Using this prior information, we are able to distinguish correctly between people and trains, since a person jumping to the track bed will always have a motion component perpendicular to the edge of the platform. Figure 3 displays a train in motion and a person jumping to the track bed region. Figure 4 displays a train in three different states.

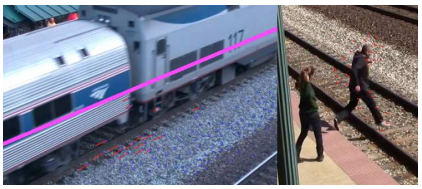


Fig. 3. A Train in Motion and a Person Jumping Off the Platform.

Fig. 4. Arriving Train, Train Discharging Passengers and a Departing Train.

To estimate motion vectors, we find the most prominent corners of each frame constrained to the track bed region [27]. To find the prominent corners we divide the region into blocks of size 5×5 and determine the eigenvalues of the covariance matrix of derivatives. Large eigenvalues (λ_1, λ_2) represent corners, salt-pepper textures, or any other pattern that can be reliably tracked. We consider a corner to be prominent if $\min(\lambda_1, \lambda_2) > \lambda$, where λ is a predefined threshold. The value of λ is set adaptively: for each block in the region we find the minimum eigenvalue, next we find the maximum value of all the minimum eigenvalues, finally we set λ to 0.01 times this maximum. Figure 5 displays how a train window contains prominent corners that can be used for tracking.

Fig. 5. Window Corners Used For Tracking.

We use a point-based tracker to track the prominent corners previously found. The Lucas Kanade Feature Tracker [28] is used from OpenCV. While the train is moving, prominent corners are displaced between frames. Displacement vectors serve as an indicator of the train speed. When the train is moving at a high speed, the displacement distance is large. In the case where a train is arriving, the displacement distance decreases until it reaches zero, indicating that the train has stopped.

2.3. Event Detection

Moving object detection is used when the track bed region is classified to be clear. The entire video frame is used and each blob in the foreground mask is assumed to belong to a person. For each blob the contours are estimated and approximated by a rectangle [29]. When the base of the rectangular contour (assumed to be the feet of a person) completely crosses the line marking the edge between the platform and the track bed, an alarm is raised indicating the detection of a person into the track bed. Figure 6 shows an example. A proximity warning is raised if 80% of the base of the rectangle contours crosses the line marking the edge between the platform and the track bed.

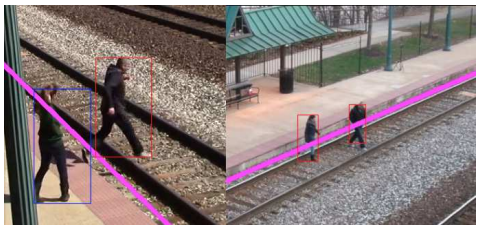


Fig. 6. Alarm Indicating a Person Crossing to the Track Bed Region.

3. EXPERIMENTAL RESULTS

3.1. The Experimental Dataset

The experimental evaluation was conducted using our new dataset. Videos were recorded from three different views where the cameras were placed at a height similar to typical surveillance cameras. Figure 7 shows three different views extracted from the dataset. Data was collected during two different days with different and changing light conditions. The dataset has a total of 60 minutes of video containing different events. Videos were recorded in High Definition (HD) resolution and then converted to Standard Definition resolution (SD, 853×480) in order to meet real-time processing requirements. Testing the proposed method was conducted using 70% of the dataset while the other 30% was used during the development stage. The original video was segmented into small clips containing the events shown in Table 1.

Fig. 7. Camera Views in the Experimental Dataset

For most of the events, we have three different synchronized views. Ground truth was manually generated using the following labels:

- "No events nor trains"
- "Person crossing to the track bed event"
- "Train arriving"

Table 1. Type of Events

Event	# of incidents
Person A jumping off the platform	4
Person B jumping off the platform	2
Person A and B jumping off the platform	2
Person A falling into the track bed	2
Person B pushing person A to the track bed	1
Train arriving	2
Train discharging passengers	3
Train departing	4

- "Train waiting or discharging passengers"
- "Train departing"

3.2. Results

A true positive (TP) is a correctly identified event, such as a person jumping off the platform, a person pushed by another person or a person falling to the track bed. A false positive (FP) is an incorrectly identified event and a false negative (FN) is a missed detection. Precision was determined and found to be 90%, while recall was 100%.

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

Train presence was correctly identified at all times. Arriving trains were detected correctly and therefore no false alarms were raised when passengers were boarding the train or being discharged. Departing trains were also correctly identified, this is essential to detecting True Positives and avoiding missed detections. Precision was found to be 90% due to a false positive. This false positive was because of misclassifying a departing train as a moving person. As the train departs and is distant from the surveillance camera, the motion vectors are not correctly estimated.

4. CONCLUSION

We presented a method to automatically detect people jumping or falling off a train platform. We detect moving objects and analyze their location in relative to the edge of the track bed. Train presence is investigated by analyzing motion vectors in the track bed region. An experimental evaluation was conducted using a video dataset that was recorded at a local train station, precision and recall were found to be 90% and 100% respectively. Train presence was correctly identified at all times. For future work, we want to investigate multi-camera scenarios, study the impact of objects moving in the immediate vicinity to the camera, estimate the depth of moving objects using motion information, and test our techniques on a lager data set.

5. REFERENCES

[1] M. Valera and S. Velastin, "Intelligent distributed surveillance systems: a review," *IEE Proceedings of Vision, Image and Signal Processing*, vol. 152, no. 2, pp. 192–204, April 2005. doi: [10.1049/ip-vis:20041147]

[2] N. Haering, P. L. Venetianer, and A. Lipton, "The evolution of video surveillance: an overview," *Machine Vision and Applications*, vol. 19, no. 5-6, pp. 279–290, October 2008. doi: [10.1007/s00138-008-0152-0]

[3] W. Li, "Anomaly detection and localization in crowded scenes," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1975–1981, June 2010. doi: [10.1109/TPAMI.2013.111] San Francisco, California.

[4] R. Mehran, A. Oyama, and M. Shah, "Abnormal crowd behavior detection using social force model," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 935–942, June 2009. doi: [10.1109/CVPR.2009.5206641] Miami, Florida.

[5] K. Smith, P. Quelhas, and D. Gatica-Perez, "Detecting abandoned luggage items in a public space," *Proceedings of the 9th IEEE International Workshop on Performance Evaluation in Tracking and Surveillance (PETS'06)*, pp. 75–82, June 2006, New York, New York.

[6] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *ACM Journal of Computing Surveys*, vol. 43, no. 3, pp. 1–43, April 2011.

[7] J. K. Aggarwal and Q. Cai, "Human motion analysis: A review," *Computer Vision and Image Understanding*, vol. 73, no. 3, pp. 428–440, March 1999.

[8] P. Turaga, R. Chellappa, V. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1473–1488, November 2008.

[9] J. Candamo, M. Shreve, D. B. Goldgof, D. B. Sapper, and R. Kasturi, "Understanding transit scenes: A survey on human behavior-recognition algorithms," *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, vol. 11, pp. 206–224, October 2010. doi: [10.1109/TITS.2009.2030963] Funchal, Madeira Island, Portugal.

[10] S. Velastin, B. A. Boghossian, and M. A. Vicencio-Silva, "A motion-based image processing system for detecting potentially dangerous situations in underground railway stations," *Transportation Research Part C: Emerging Technologies*, vol. 14, pp. 96–113, October 2006. doi: [10.1016/j.trc.2006.05.006]

[11] "Public Transportation Network in Paris," URL: http://www.ratp.fr.

[12] M. Spirito, C. S. Regazzoni, and L. Marcenaro, "Automatic detection of dangerous events for underground surveillance," *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 195–200, September 2005. doi: [10.1109/AVSS.2005.1577266] Como, Italy.

[13] J. Black, S. Velastin, and B. Boghossian, "A real time surveillance system for metropolitan railways," *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 189–194, September 2005. doi: [10.1109/AVSS.2005.1577265] Como, Italy.

[14] C. Seyve, "Metro railway security algorithms with real world experience adapted to the ratp dataset," *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 177–182, September 2005. doi: [10.1109/AVSS.2005.1577263] Como, Italy.

[15] K. Schwerdt, D.Maman, P. Bernas, and E. Paul, "Target segmentation and event detection at video-rate: the eagle project," *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 183–188, September 2005. doi: [10.1109/AVSS.2005.1577264] Como, Italy.

[16] A. Monitzer, "Using video surveillance to detect dangerous situations in underground stations by computer vision," *Proceedings of Scientific Presentation and Communication*, 2006, URL:http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.123.5657.

[17] B. Krausz and R. Herpers, "Metrosurv: detecting events in subway stations," *Multimedia Tools and Applications*, vol. 50, no. 1, pp. 123–147, October 2010. doi: [10.1007/s11042-009-0367-8]

[18] Y. Park and D. Lee, "3d vision-based security monitoring for railroad stations," *Journal of the Optical Society of Korea*, vol. 14, no. 4, pp. 451–457, December 2010. doi: [0.3807/JOSK.2010.14.4.451]

[19] J. Schutte and S. Scholz, "A new security and safety solution for public guided transport," *Proceedings on Joint Rail*, pp. 245–250, March 2009. doi: [10.1115/JRC2009-63015] Pueblo, Colorado.

[20] F. Ziliani, S. Velastin, F. Porikli, L. Marcenaro, T. Kelliher, A. Cavallaro, and P. Bruneaut, "Performance evaluation of event detection solutions: the creds experience," *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 201–206, September 2005. doi: [10.1109/AVSS.2005.1577267] Como, Italy.

[21] A. Cavallaro and F. Ziliani, "Characterisation of tracking performance," *Proceedings of Workshop on Image Analysis For Multimedia Interactive Services (WIAMIS)*, pp. 13–15, April 2005, Montreux, Switzerland.

[22] "Open Source Computer Vision Library," URL: http://www.opencv.org.

[23] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, vol. 2, pp. 28–31, August 2004. doi: [10.1109/ICPR.2004.1333992] Cambridge, England.

[24] Z. Zivkovic and F. Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern recognition letters*, vol. 27, no. 7, pp. 773–780, May 2006. doi: [10.1016/j.patrec.2005.11.005]

[25] J. Banfield and A. Raftery, "Model-based gaussian and non-gaussian clustering," *Biometrics*, vol. 49, no. 3, pp. 803–821, September 1993.

[26] C. Ronse and H. J. A. M. Heijmans, "The algebraic basis of mathematical morphology: Ii. openings and closings," *Computer Vision Graphics and Image Processing (CVGIP): Image Understanding*, vol. 54, no. 1, pp. 74–97, July 1991. doi: [10.1016/1049-9660(91)90076-2]

[27] J. Shi and C. Tomasi, "Good features to track," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (CVPR)*, pp. 593–600, June 1994. doi: [10.1109/CVPR.1994.323794] Seattle, Washington.

[28] J. Y. Bouguet, "Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm," *Intel Corporation*, vol. 2, p. 3, 2001.

[29] S. Suzuki and K. Abe, "Topological structural analysis of digitized binary images by border following," *Computer Vision, Graphics, and Image Processing*, vol. 30, no. 1, pp. 32–46, April 1985. doi: [10.1016/0734-189X(85)90016-7]

Characterizing The Uncertainty of Classification Methods and Its Impact on the Performance of Crowdsourcing

Javier Ribera, Khalid Tahboub, and Edward J. Delp

Video and Image Processing Laboratory (VIPER)
School of Electrical and Computer Engineering
Purdue University, West Lafayette, Indiana USA

ABSTRACT

Video surveillance systems are widely deployed for public safety. Real-time monitoring and alerting are some of the key requirements for building an intelligent video surveillance system. Real-life settings introduce many challenges that can impact the performance of real-time video analytics. Video analytics are desired to be resilient to adverse and changing scenarios. In this paper we present various approaches to characterize the uncertainty of a classifier and incorporate crowdsourcing at the times when the method is uncertain about making a particular decision. Incorporating crowdsourcing when a real-time video analytic method is uncertain about making a particular decision is known as online active learning from crowds. We evaluate our proposed approach by testing a method we developed previously for crowd flow estimation. We present three different approaches to characterize the uncertainty of the classifier in the automatic crowd flow estimation method and test them by introducing video quality degradations. Criteria to aggregate crowdsourcing results are also proposed and evaluated. An experimental evaluation is conducted using a publicly available dataset.

Keywords: video surveillance, online active learning, crowdsourcing, crowd flow, video analytics

1. INTRODUCTION

Video surveillance systems provide a wide range of capabilities for security purposes. Commercial, military and law enforcements applications demand intelligent video surveillance systems capable of understanding the scene being recorded [1], [2], [3], [4], [5]. Real-life settings introduce many challenges that can impact the performance of real-time video analytics. Video analytics are desired to be resilient to adverse and changing scenarios. However, maintaining an acceptable level of accuracy is often a difficult task in a changing set of scenarios. Occlusions, shadows, sudden and gradual environmental conditions, and distortions in the video due to packet losses or compression are important challenges that analytic techniques must face.

"Crowdsourcing," has been shown to be an effective solution to deal with tasks that require human comprehension. Originally, crowdsourcing was conceived as a way to obtain services from a crowd of unspecified users [6]. Recently, a wider and consistent definition of crowdsourcing was presented in [7]. The concept of the of "wisdom of the crowd" leads to an expectation of an accurate outcome if the crowd fulfills some criteria such as independence and diversity [8]. An extensive study reviewing the current crowdsourcing systems on the world-wide web is described in [9]. Several commercial platforms already exist in order to reach the public crowd, ask for contributions and gather results. Prominent examples are Amazon's Mechanical Turk (MTurk) [10], Mob4hire [11], uTest [12] or Freelancer [13]. Issues concerning using commercial crowdsourcing systems for law enforcement arise, due to requirements such as video contents protection [14]. In [15], we presented a web-based tool was developed to allow fine control over annotations of surveillance videos.

Some authors have proposed to use human intelligence to assist machines in automated tasks. In [16], autonomous robots apprehend new objects using machine learning. The new objects are learned by crowdsourced labels obtained through MTurk. In [17], crowdsourcing is used to refine the model of the classifier of an object detector, by iteratively identifying unlabeled data and requesting its labeling through MTurk. In [18], an active learning approach is proposed to consider multiple annotators with varying expertise to label the same data

This work was partially supported by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0001. Address all correspondence to Edward J. Delp, ace@ecn.purdue.edu

sample. In [19], active learning is used to collect on-demand labels for a human activity recognition application. These labels are collected from multiple workers annotating the same label set.

In this paper, crowdsourcing is utilized to help video surveillance systems perform better, as proposed in [20]. Thus, video analytics benefit from human intelligence to increase their accuracy. We evaluate our proposed approach by testing a method we developed previously for crowd flow estimation. Crowd flow estimation refers to the number of people crossing a specific region in a given period of time [21]. To avoid confusion, we shall refer to the people assisting the automatic analysis perform better as the "o-crowd." The method reaches out to the o-crowd when it is uncertain about making a particular decision. Different methods to characterize the uncertainty of the classifier are discussed and evaluated. In this paper, the method reaches out to more than one o-crowd member and aggregate results accordingly. We propose two different approaches to aggregate the contributions of the o-crowd. We conduct an experimental evaluation using a publicly available dataset.

2. ONLINE ACTIVE LEARNING FROM CROWDS

Active learning has been widely used to allow methods to learn by querying users data points of interest [22]. In a context where unlabeled data is abundant, but labeling is expensive, minimizing the amount of labeling is crucial. By letting the method dynamically select the data to be queried, the most useful points can be inquired in an optimal way. With an active learning approach, intelligent video surveillance systems would require no manual intervention to select the points to be queried.

Active learning can be split into two distinct categories: pool-based active learning and online active learning [18]. In a pool-based active learning, a batch of points are delivered to the classifier and the optimal points are selected. The classifier has the flexibility to select between a pool of data points while in an online active learning, the classifier must decide in real-time if a new data point has to be labeled or not [18].

In this work we employ online active learning. The automatic crowd flow estimation method analyzes frame by frame the input video and quantizes the uncertainty of its classification. When this uncertainty is above a given threshold, the uncertain frame is queried to the crowdsourcing members. The aggregated answers are used to label that uncertain point. Finally, this labeled point is used to train the classifier to enhance future classifications, hence increasing the final video analytics accuracy.

As will be described in the next section, crowd flow estimation relies on the estimation of the level of crowdedness. To achieve this goal, we extract features from a video frame and classify them into one of the levels of crowdedness. Sixteen scalar texture features are computed and concatenated to form a feature vector. The classifier has to classify a frame's feature vector into one of the "reference texture feature vector" in a 16-D feature space. Reference texture feature vectors are feature vectors that were labeled during offline training. We shall refer to the 16-D feature vector extracted from a video frame as a data point. We shall also refer to "reference texture feature vectors" as reference points.

Let t be the data point for a given frame and $\tau_l, l = 1, 2, 3, ..., L$ be the reference points such that τ_1 is the nearest reference point to the data point t while τ_L is the furthest. The distance from t to τ_i is denoted by d_i. Accordingly, we have $d_i < d_j, \forall i < j$. The distances from τ_i to τ_j are denoted by $d_{i,j}$. Three different methods are proposed to quantize the uncertainty of the classification of t.

The first characterization of uncertainty consists of the ratio of the distances to the two nearest reference points as shown in Equation (1). The second characterization, expressed in Equation (2), is a generalization of the first one and uses the ratio to all of the reference points. The third characterization, in Equation (3), makes use of the distance to the border d_b that separates d_1 and d_2. By definition, the three characterizations are bounded between 0 and 1, as any distance must be positive.

$$\mu_1 = \frac{d_1}{d_2} \tag{1}$$

$$\mu_2 = \sum_{i=2}^{L-1} \frac{d_1}{d_i} \tag{2}$$

$$\mu_3 = \frac{1}{1 + \frac{d_b}{d_{12}}} \tag{3}$$

If the uncertainty μ exceeds a threshold denoted by α, the data point t is considered uncertain and is referred to the o-crowd for labeling. Hence, α represents the maximum uncertainty allowed in the system. This threshold defines "certainty areas" around the reference points. When a data point falls inside the certainty area it will be considered "certain," it is not referred to the o-crowd and is classified as the same level as the nearest reference point. Data points outside all certainty areas are referred to the o-crowd for its labeling and also used as a future reference point. Figures 1 and 2 depict the certainty areas for the first and third uncertainty characterization.

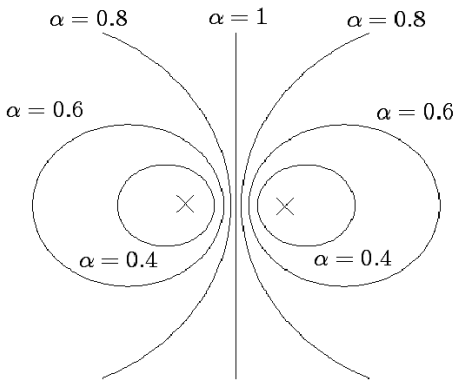


Figure 1. Uncertainty characterization #1. Certainty areas around two reference points corresponding to four values of α. They correspond to hyperspheres, whose size increases with α.

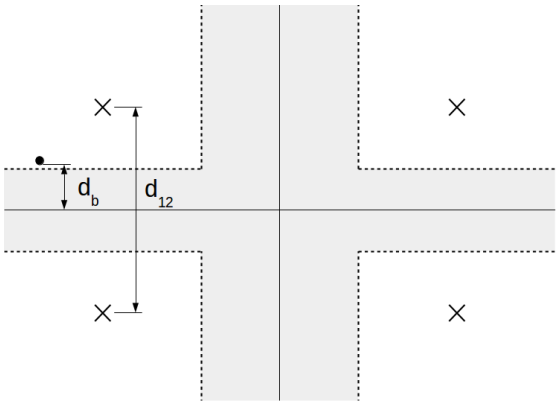


Figure 2. Uncertainty characterization #3. Certainty areas around four reference points corresponding to four values of α. The uncertainty area (in gray) is a region in the margin around the border in between two references. The width of this region is relative to the separation of the two references and α.

By tuning α, the size of the certainty area can be controlled. A smaller certainty area implies a lower probability for a testing data point falling inside a certainty area. Therefor, we denote α as the "crowdsourcing parameter," as it is related to how often we reach out to the o-crowd. When we reach out to the o-crowd for the labeling of a data point, we obtain answers from several o-crowd members. Labels from the o-crowd can be aggregated in different ways. Crowd members vary in their abilities and performance history. We investigate two different approaches to aggregate results. The first approach averages all of the o-crowd answers while the second approach calculates the average of the subset with the smallest variance.

3. CROWD FLOW ESTIMATION

We evaluate our proposed approach for online active learning from crowds by testing a method we developed previously for crowd flow estimation [21]. Crowd flow is defined as the number of people crossing a specific region of time in a given period of time. A direct approach would imply detecting and tracking every individual entering and leaving the scene. Lack of scalability and poor robustness arise when trying to track simultaneously hundreds of targets. This method follows an indirect approach, where the characteristics of the crowd are assumed to be related to low level features of the image, such as the foreground segmentation and texture features.

Two regions must be defined by the operator of the system: the Tripwire and the Region of Interest (ROI). The Tripwire is the region through which people crossing are counted, and the ROI is a larger area surrounding the Tripwire. Figure 3 shows an example of a Tripwire and a ROI.

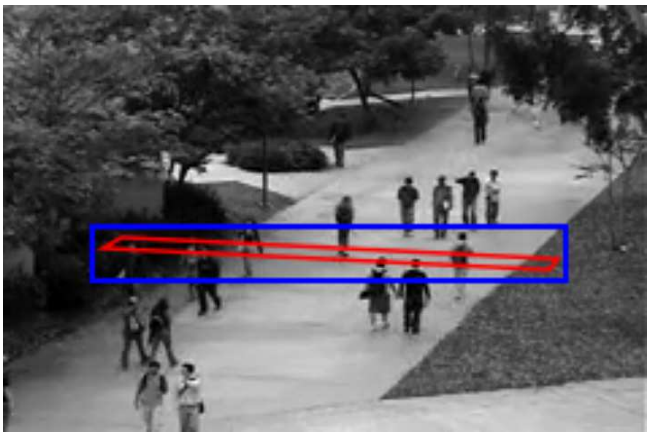


Figure 3. Example of a Region of Interest (ROI) and a Tripwire drawn over a frame of a surveillance video. ROI is in blue, and Tripwire in red.

We estimate the crowd flow by scaling the foreground pixels in the Tripwire according to the following equation:

$$v_N = \frac{S_N}{C} \tag{4}$$

where v_N represents the number of people that have crossed the Tripwire up to the frame number N. S_N represents the accumulation of foreground pixels during that period. C is the factor that relates both the number of number foreground pixels and the people count. Therefore, it denotes the number of visible foreground pixel per person. We shall refer to C as the scaling factor.

This proportionality rule holds true as long as the level of occlusion remains constant. Occlusion is related to crowdedness, as more crowdedness means that less pixels that correspond to a person can be seen. Thus, the value of C depends on the level of crowdedness of the scene. The scaling factor depends on the crowdedness of the frame, resulting in Equation (5).

$$v_N = \sum_{n=1}^{N} \frac{\sum_{x,y \epsilon \Re} I_n(x,y)}{C_n} \tag{5}$$

where the scaling factor C_n depends on the crowdedness of frame number n. The Tripwire region is denoted by $\Re$. $I_n(x, y)$ is the foreground mask of the current frame number n. $I_n(x, y)$ is an indicative function that is 1 when the pixel (x, y) belongs to the foreground segmentation and 1 otherwise, i.e,

$$I_n(x,y) = \begin{cases} 1 & : (x,y) \in \text{foreground segmentation} \\ 0 & : (x,y) \notin \text{foreground segmentation} \end{cases} \tag{6}$$

The foreground mask is computed using the moving object detection method proposed in [23], [24]. Only the accumulation of foreground pixels in the Tripwire is needed. Note that the computational cost of computing Background Subtraction in a small region is much lower than in the whole frame. Figure 4 shows an example of the result of the foreground segmentation on a Tripwire.

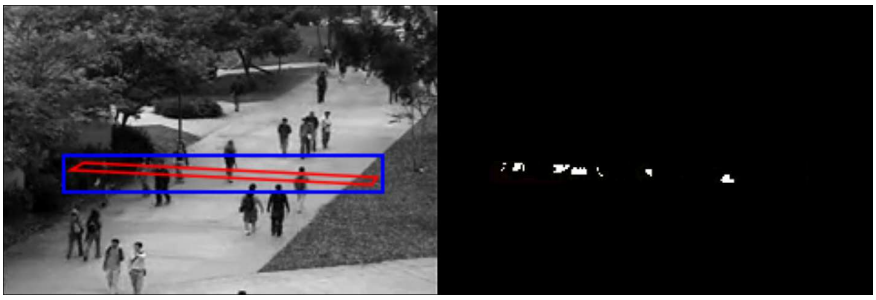


Figure 4. An arbitrary frame, on the left, and the foreground mask of the Tripwire on the right.

In [20], the foreground segmentation is weighted with a weighting function $\omega(x, y)$ that scales the contribution to the foreground mask of every pixel (x, y). This intends to take into account perspective distortions that cause objects closer to the camera to contribute with more foreground pixels. In our experiments there were no significant differences in the results, as in our experiments all object are approximately at the same distance to the camera. Consequently, we set $\omega(x, y) = 1, \forall(x, y) \in \Re$.

3.1 Crowdedness Estimation

When the foreground pixel count S_n of a frame is computed it must be scaled properly with the correct scaling factor $1/C$. In order to compute C, the crowdedness of the frame must be estimated. As C is the number of visible foreground pixels per person, a higher level of crowdedness should correspond to a lower value of C.

This method uses the approach described in [25], [26], where the crowdedness is related to the texture of the image. As Figure 5 shows, a sparse scene presents a fine texture, while a crowded scene presents a coarse texture. In [27], various methods to characterize the texture of an image are described. In this method, the texture of a scene is characterized by means of a Gray Level Co-occurrence Matrix (GLCM) [28], [27]. The GLCM matrix models a texture by identifying the probability that a pixel with a given gray level is adjacent to another specific gray level.

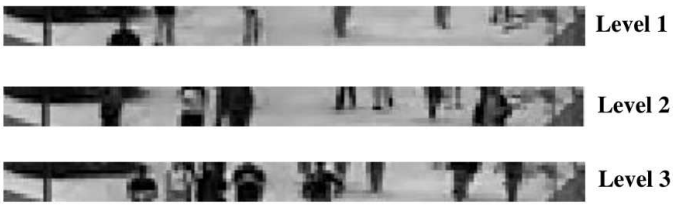


Figure 5. A sparse scene, with a low level of crowdedness, presents a fine texture. In contrast, a crowded scene, with a high level of crowdedness, presents a coarse texture.

As we only need to scale the foreground pixel count inside the Tripwire, the crowdedness estimation has to be done only in the Region of Interest. Thus, the ROI must surround the Tripwire to represent its level of

crowdedness. We create four GLCM matrixes from the ROI. Each matrix is computed using different directions to consider adjacent pixels right (0), top-right (45), top (90), and top-left (135). From each matrix, 4 scalars are extracted: energy, entropy, homogeneity and contrast. The expressions to compute each scalar correspond to Equations (7), (8), (9) and (10).

$$Energy(P) = \sqrt{\sum_{i,j} p_{ij}^2} \tag{7}$$

$$Entropy(P) = -\sum_{i,j} p_{ij} \log p_{ij} \tag{8}$$

$$Homogeneity(P) = \sum_{i,j} \frac{p_{ij}}{1 + (i-j)^2} \tag{9}$$

$$Contrast(P) = \sum_{i,j} p_{ij}(i-j)^2 \tag{10}$$

With these 16 scalar features, a 16-D texture feature vector, t_n, is assembled to represent the texture feature of the ROI of the frame number n. t_n is classified into one of the "reference texture feature vectors" denoted as $\tau_l, l = 1, 2, 3, ..., L$. Reference texture feature vectors are texture feature vectors that are computed during the training stage by an expert member of the "o-crowd." Their corresponding scaling factor C is also computed during the training process. L is the number of crowdedness levels manually specified in the training stage. The classifier assigns t_n to one of the τ_l by finding the nearest neighbor. Then, the classification problem results in a K-N Nearest Neighbors in which $K = 1$.

3.2 Training

The classifier must be initially trained in order to classify the 16-D feature vector t into one of the $\tau_1, \tau_2, ..., \tau_L$. This training process is conducted by an expert o-crowd member. The first output of the training data are the L reference texture feature vectors. Each vector represents the texture of one of the L levels of crowdedness. As mentioned earlier, the scaling factor C relating the number of foreground pixels to the number of people crossing the Tripwire region is dependent on the level of crowdedness. The second output is the L scaling factors $C_1, C_2, ..., C_L$ that correspond to each crowdedness level.

The training process is conducted as follows: first, the system operator is asked to mark the Tripwire and Region Of Interest. Next, M video frames are chosen randomly from the training video segment. Second, the operator manually classifies each frame into one of the L levels of crowdedness. The reference vectors τ_l are computed as the average of all the feature vectors of the frames classified as the same level of crowdedness. Finally, the scaling factors of each crowdedness level must be estimated. We find the longest stable period for each level of crowdedness. A stable period is defined as a set of consecutive frames with the same level of crowdedness. These L video segments, containing the stable periods, are shown to the operator and the number of people crossing the Tripwire is asked. The scaling factor C of each level of crowdedness is determined according to Equation (4).

4. EXPERIMENTAL RESULTS

4.1 Testing

For our experiments we used the following publicly available surveillance video dataset: University of California, San Diego (UCSD) pedestrian dataset [29]. It consists of a 54 minutes long video, containing $L = 3$ distinct crowdedness levels. The video was segmented in the following way: the first 30 minutes were used solely for training, and the remaining minutes were used to create 5 consecutive clips of 4 minutes length. These 5 video segments were used for testing purposes.

Enhancing the automatic crowd flow estimation using crowdsourcing was shown to improve its performance in [20], showing resilience to degradations in the video quality. Crowdsourcing was helpful to make the system

	α	Average error rate
Original quality	1	8.1%
Degraded quality	1	31.7%

Table 1. The average error rate when the video quality is degraded increases from 8 to 32%.

adapt to compression and contrast changes that may come from network congestion. In [20], the size of the o-crowd consisted of two expert members. Both contributions from the o-crowd members were averaged . In this work, we compare the two proposed approaches to characterize the uncertainty of the classifier. We also test a generic o-crowd consisting of 5 members. We investigate two different approaches to aggregate results. The first approach averages all of the 5 o-crowd answers while the second approach calculates the average of the subset ($K = 3$) with the smallest variance.

The threshold used in the foreground segmentation was set empirically to 50. This threshold represents the distance from a pixel value to the background model to decide whether it belongs to the foreground or the background. The learning rate, that represents how fast the background model is updated, was left to be decided and updated automatically by the moving object detection presented in [23], [24] and implemented in OpenCV 2.4.9 [30]. The GLCM matrix was computed using the scikit-image library [31] and using 8 gray levels. The number of random snapshots of the ROI used to compute the reference texture vectors during the training stage was $M = 50$.

We measured the crowd flow error rate and the utilization of the o-crowd in every segment. The error rate is defined as the difference between the estimated and the ground truth, as a percentage of the ground truth. The utilization of the o-crowd is the number of tasks which have been sent to the o-crowd, which corresponds to the number of texture feature vectors that have appeared outside the certainty areas.

4.2 Results and Discussion

The UCSD dataset does not include ground truth information. Therefore, we used our best judgment to provide the ground truth data. As video segments may start or end while some people are crossing the Tripwire, the ground truth can be "± 0.5" away from the true value. We made the video segments as long as possible to minimize the impact of this fact on the measure of the accuracy.

The degradation of the testing segments consisted of H.264 compression using ffmpeg [32] and a boost in the contrast of the videos. The Constant Rate Factor (CRF) was set to 35 and the contrast increased to 2.68 using the eq2 filter. As a result, the size of the degraded video files was reduced to a 40%.

The average error rate with no crowdsourcing is shown in Table 1. The average error rate is the average of the error rates in all the segments. The error rate increased by a factor of 4 when the video quality was degraded.

In Table 2, we compare the resulting error rate when two uncertainty characterizations are used. While using the second type of characterization, the resulting value of μ_2 is always smaller than μ_1. To make it a fair comparison we chose different values for α for each approach such that the resulting total utilization is similar. The total utilization is the total number of times we reach out to the o-crowd for the labeling of a data point. Table 2 results were computed by using a single expert o-crowd member. The second type of characterization achieves a lower error rate. The results of the third characterization are not included in Table 2. We experimentally observed that even after degradation all the data points never fell close to the border, which implies a low value for μ_3. For example, in the first segment, the value of μ_3 was 0.2, which implies that the distance to the border d_b was 4 times the distance between the 2 closest reference vectors, d_{12}. As a result, we conclude that the estimation in the degraded video of this dataset can not be enhanced with this type of uncertainty characterization.

Table 3 presents the results for the two approaches to aggregate results for 5 o-crowd members using the first uncertainty characterization. The first row shows the results when all the answers of the 5 o-crowd members are averaged. The second row shows the results when the average of the subset ($K = 3$) with the smallest variance is calculated. In this experiment, the first characterization of the uncertainty was used with $\mu_1 = 0.6$.

	Uncertainty characterization	α	Total utilization	Average error rate
Degraded quality	μ_1	0.6	13	15.2%
Degraded quality	μ_2	0.5	13	12.5%

Table 2. The error rate when using the second uncertainty characterization was lower than using the first characterization, given the same amount of labeling necessary.

Aggregation of K o-crowd members	Total utilization	Average error rate
K=5	13	26%
K=3	13	19.5%

Table 3. By aggregating only the $K = 3$ o-crowd members with the most similar labels, we achieve more reduction in the error rate.

As the method is expected to learn from crowdsourcing, the certainty areas should evolve so that less number of testing data points fall outside these regions. As the o-crowd is used when a testing data point is uncertain, a decreasing utilization of the o-crowd would indicate proper learning of the method. This behavior is depicted in Figure 6.

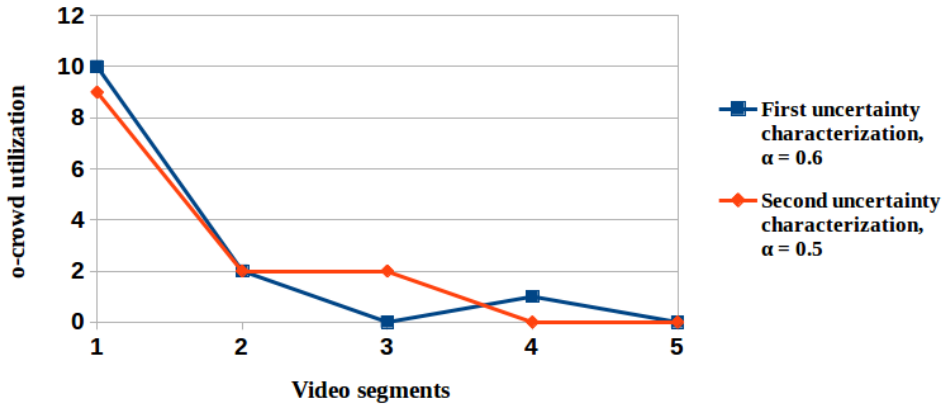


Figure 6. The system learns from the crowdsourcing input, so the utilization of the o-crowd keeps decreasing throughout time.

5. CONCLUSION AND FUTURE WORK

In this paper, we proposed different characterizations of the uncertainty of a classification. They were used in a video surveillance application, a crowd flow estimation method enhanced by crowdsourcing. The method manages to identify the uncertainties properly. Loss in accuracy due to degradation in the video quality is addressed by the incorporation of crowdsourcing to achieve active learning. An online active learning scheme is used, where the uncertain data points to be labeled by the o-crowd members are selected in a frame-by-frame fashion. Various characterizations of the uncertainty of the classifier are proposed and compared. Aggregating results from a subset of the o-crowd with the smallest variance is shown to yield better results compared to averaging the results fro the whole o-crowd. In future, we will incorporate the performance history of o-crowd members in to the model by adapting a pyramid model to differentiate the abilities and skills of different o-crowd members.

REFERENCES

[1] K. Yang, E. J. Delp, and E. Du, "Categorization-based two-stage pedestrian detection system for naturalistic driving data," *Signal, Image and Video Processing*, vol. 18, no. 1, pp. 135–144, October 2014.

[2] S. Srivastava and E. J. Delp, "Video-based real time surveillance of vehicles," *Journal of Electronic Imaging*, vol. 22, no. 4, pp. 041 103(1)–(16), December 2013.

[3] S. Srivastava and E. J. Delp, "Standoff video analysis for the detection of security anomalies in vehicles," *Proceedings of IEEE Applied Imagery Pattern Recognition*, pp. 1–8, October 2010, Washington, DC.

[4] S. Srivastava, K. K. Ng, and E. J. Delp, "Co-ordinate mapping and analysis of vehicle trajectory for anomaly detection," *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 1–6, July 2011, Barcelona, Spain.

[5] B. Delgado, K. Tahboub, and E. J. Delp, "Automatic detection of abnormal human events of train platforms," *Proceedings of the IEEE National Aerospace & Electronics Conference*, June 2014, Dayton, OH.

[6] J. Howe, "The rise of crowdsourcing," *Wired magazine*, vol. 14, no. 6, pp. 1–4, 2006, Dorsey Press.

[7] E. Estellés-Arolas and F. González-Ladrón-De-Guevara, "Towards an integrated crowdsourcing definition," *Journal of Information Science*, vol. 38, no. 2, pp. 189–200, April 2012.

[8] J. Surowiecki, *The Wisdom of Crowds*. Knopf Doubleday Publishing Group, 2005.

[9] A. Doan, R. Ramakrishnan, and A. Y. Halevy, "Crowdsourcing systems on the world-wide web," *Communications of the ACM*, vol. 54, no. 4, pp. 86–96, 2011.

[10] "Amazon Mechanical turk," URL: https://www.mturk.com.

[11] "Mob4hire," URL: http://www.mob4hire.com.

[12] "uTest," URL: https://www.utest.com.

[13] "Freelancer," URL: https://www.freelancer.com.

[14] D. Brabham, *Crowdsourcing*. The MIT Press, 2013.

[15] N. J. Gadgil, K. Tahboub, D. Kirsh, and E. J. Delp, "A web-based video annotation system for crowdsourcing surveillance videos," *Proceedings of the SPIE/IS&T Electronic Imaging, Imaging and Multimedia Analytics in a Web and Mobile World*, vol. 9027, pp. 90 270A–1–12, March 2014, San Francisco, CA.

[16] A. Sorokin, D. Berenson, S. Srinivasa, and M. Hebert, "People helping robots helping people: Crowdsourcing for grasping novel objects," *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2117–2122, October 2010, Taipei, Taiwan.

[17] S. Vijayanarasimhan and K. Grauman, "Large-scale live active learning: Training object detectors with crawled data and crowds," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1449–1456, June 2011, Providence, RI.

[18] Y. Yan, R. Rosales, G. Fung, and J. Dy, "Active learning from crowds," *Proceedings of the International Conference on Machine Learning*, pp. 1161–1168, June 2011, Bellevue, WA.

[19] W. S. Lasecki, Y. C. Song, H. Kautz, and J. P. Bigham, "Real-time crowd labeling for deployable activity recognition," pp. 1203–1212, February 2013, San Antonio, Texas.

[20] J. Ribera, K. Tahboub, and E. J. Delp, "Automated crowd flow estimation enhanced by crowdsourcing," *Proceedings of the IEEE National Aerospace & Electronics Conference*, June 2014, Dayton, OH.

[21] S. Srivastava, K. K. Ng, and E. J. Delp, "Crowd flow estimation using multiple visual features for scenes with changing crowd densities," *Proceedings of the 8th IEEE International Conference on Advanced Video and Signal-Based Surveillance*, pp. 60–65, August-September 2011, Klagenfurt, Austria.

[22] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models," *arXiv preprint cs/9603104*, 1996.

[23] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," *Proceedings of the International Conference on Pattern Recognition*, vol. 2, pp. 28–31, August 2004.

[24] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern recognition letters*, vol. 27, no. 7, pp. 773–780, May 2006.

[25] A. N. Marana, S. A. Velastin, L. F. Costa, and R. A. Lotufo, "Automatic estimation of crowd density using texture," *Safety Science*, vol. 28, no. 3, pp. 165–175, April 1998.

[26] A. N. Marana and S. A. Velastin and L. F. Costa and R. A. Lotufo, "Estimation of crowd density using image processing," *Proceedings of the IEE Colloquium on Image Processing for Security Applications*, pp. 11/1–11/8, March 1997, London, United Kingdom.

[27] R. M. Haralick, "Statistical and structural approaches to texture," *Proceedings of the IEEE*, vol. 67, no. 5, pp. 786–804, May 1979.

[28] R. M. Haralick, K. Shanmugam, and H. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man and Cybernetics*, no. 6, pp. 610–621, November 1973.

[29] A. B. Chan and N. Vasconcelos, "Modeling, clustering, and segmenting video with mixtures of dynamic textures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 5, pp. 909–926, May 2008.

[30] "OpenCV," URL: http://www.opencv.org.

[31] "scikit-image," URL: http://scikit-image.org.

[32] "FFmpeg," URL: http://www.ffmpeg.org.

Efficient Graph-Cut Tattoo Segmentation

Joonsoo Kim, Albert Parra, He Li, Edward J. Delp

Video and Image Processing Lab *(VIPER)*
School of Electrical and Computer Engineering
Purdue University
West Lafayette, Indiana, USA

ABSTRACT

Law enforcement is interested in exploiting tattoos as an information source to identify, track and prevent gang-related crimes. Many tattoo image retrieval systems have been described. In a retrieval system tattoo segmentation is an important step for retrieval accuracy since segmentation removes background information in a tattoo image. Existing segmentation methods do not extract the tattoo very well when the background includes textures and color similar to skin tones. In this paper we describe a tattoo segmentation approach by determining skin pixels in regions near the tattoo. In these regions graph-cut segmentation using a skin color model and a visual saliency map is used to find skin pixels. After segmentation we determine which set of skin pixels are connected with each other that form a closed contour including a tattoo. The regions surrounded by the closed contours are considered tattoo regions. Our method segments tattoos well when the background includes textures and color similar to skin.

Keywords: tattoo segmentation, graph-cut segmentation, skin color model, visual saliency map, closed contour

1. INTRODUCTION

A large percentage of criminal gang members use tattoos to show gang affiliation and to draw attention to events related to criminal activity. For this reason law enforcement is interested in exploiting tattoos as an information source to identify, track and prevent gang-related crimes. The use of tattoos for person identification and tattoo symbol interpretation is an interesting problem for law enforcement. Several tattoo image retrieval systems, that retrieve similar tattoo images from a tattoo image database, have been proposed [1]–[9]. Most of these methods manually crop the tattoo image to remove varying background from an image, which is an important preprocessing step to improve image retrieval accuracy. In [1], [2], [5], [7]–[9] tattoo segmentation was used to extract a tattoo shape as well as to remove varying background from an image. Tattoo segmentation is not an easy problem because existing segmentation methods do not extract only the tattoo when the background includes textures. In addition, existing skin detection methods used for tattoo segmentation are not robust when the background has color similar to skin tones, there is an illumination change, and skin tones of other races should be detected.

There exist many methods that use tattoo segmentation for content-based tattoo image retrieval. In [7], [9] a Sobel operator and morphological operators are used to extract low level features such as color, texture and shape. However, all the images are pre-cropped and it is assumed that the background is mostly skin and homogeneous. Most tattoo images have more complex backgrounds and include many different textures. Instead of using a morphological operator, which is not robust to weak edges, active contour based segmentation (snakes) is used in [9]. This method also uses pre-cropped images with skin background. Since the segmentation methods used in [7]–[9] are very sensitive to textures they often fail to separate the tattoo from complex backgrounds. In [1], [2], [5] efficient tattoo segmentation methods are introduced for non pre-cropped images. The use of the

This work was supported by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0001. Address all correspondence to Edward Delp (ace@ecn.purdue.edu). Some of the tattoo images shown in this paper were obtained from the Indiana Gang Network (INGangNetwork). We gratefully acknowledge their contribution.

HSV color space to find skin regions is used to segment tattoos in [1]. However, the fixed color range defined to detect skin pixels is restricted to Asian people, and it can be affected by illumination, background, and camera characteristics. In [5] tattoo segmentation is done using skin detection followed by a figure-ground segmentation. Based on the assumption that the center region of the image contains skin both the regions including skin and tattoo are detected by merging the regions connected with the center region. Then, a figure-ground segmentation using k-means clustering was used in the RGB color space to distinguish a tattoo region from a skin region. The assumption is not true in general and k-means clustering in the color space cannot distinguish skin and tattoo when a tattoo has various colors. In [2] a visual saliency map model is used along with Grabcut [10] and QCC (Quasi Connected Components) for tattoo segmentation. First, the region around the tattoo is detected based on a visual saliency model. Then, Grabcut segmentation is used in the region. Finally, the tattoo region is segmented using the result of Grabcut with QCC. However, the accuracy of this segmentation method highly depends on the accuracy of a visual saliency map.

Our contributions in this paper are that we limit the regions for graph-cut segmentation to the regions near image edges and use a variation of graph-cut segmentation with a skin model and a saliency map. While a saliency map and graph-cut based segmentation are used in [2], our method is different in that we only consider regions near image edges for the regions where the graph-cut based segmentation is used and we use a statistical skin model. By combining a skin model and a saliency model, our method can detect tattoo regions more accurately when the saliency model fails to find the regions.

2. PROPOSED METHOD

We define the tattoo segmentation problem as finding skin pixels around a tattoo assuming a tattoo is surrounded by skin. Therefore, we do not need to segment all pixels in the image. The regions near image edges are the only ones considered for segmentation (there are always edges between tattoo and skin). We call these regions "possible segmentation regions." We detect skin pixels in the regions near image edges using a probabilistic skin color model based a Gaussian Mixture Model. If there are regions that have color similar to skin in the background the skin color model fails to detect the skin around the tattoo. We additionally use a visual saliency map to focus on skin near tattoo regions. Our problem can be then considered as a 2-class labeling problem to classify each pixel around edges as skin or non-skin. A variation of graph-cut segmentation [11] using a skin color model and a visual saliency map is used in our proposed system. After the segmentation we check which set of skin pixels are connected with each other that forms a closed contour including a tattoo. The regions surrounded by the closed contours are considered tattoo regions. A block diagram of our proposed tattoo image segmentation system is shown in Figure 1.

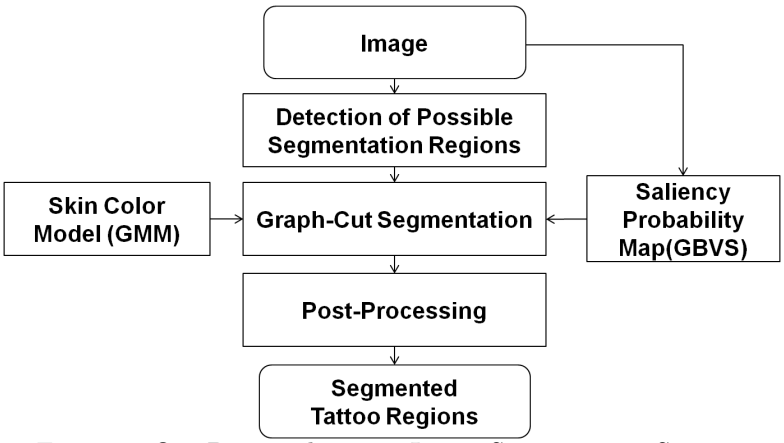
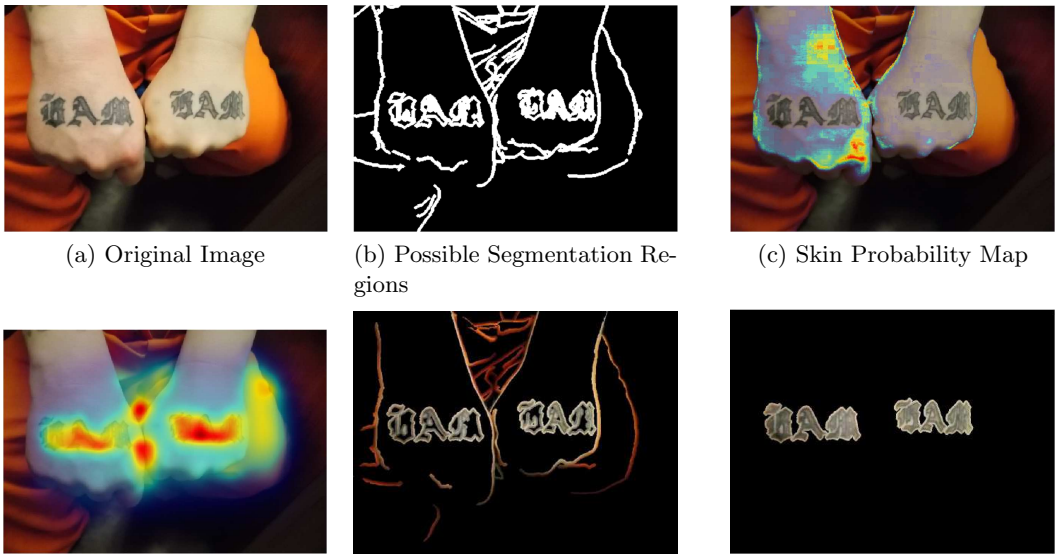


Figure 1: Our Proposed Tattoo Image Segmentation System.

2.1 Detection of Segmentation Regions

In this section we describe how the possible segmentation regions are detected. Since we only focus on regions near edges edge detection is done first. The Canny edge detector is used to find edges for each RGB color channel

separately. The edge regions from each channel are then combined. Morphological dilation is then used to the combined the edge regions to find the regions near edges. These regions are considered as possible segmentation regions. In Figure 2b white regions are possible segmentation regions. There are two advantages to limiting the regions for segmentation. First, the segmentation errors outside of the possible segmentation regions can be avoided. Second, the segmentation execution time can be reduced.



(a) Original Image (b) Possible Segmentation Regions (c) Skin Probability Map

(d) Visual Saliency Map (e) Graph-Cut Segmentation (f) Tattoo Segmentation

Figure 2: Overall Segmentation Process.

2.2 Graph-Cut Segmentation

Once the regions for segmentation are determined, we find skin pixels by using a 2-class labeling approach. Graph-cut segmentation [11] is described by the Gibbs energy:

$$E(x) = \sum_{i \in I} D(x_i) + \lambda \sum_{i \in I, j \in N_i} V(x_i, x_j), \tag{1}$$

where i is a pixel, I is an image, N_i is neighborhood pixels of pixel i, $x_i \in \{0(background), 1(foreground)\}$, $D(x_i)$ is the data term, and $V(x_i, x_j)$ is the smoothness term. To construct the graph for this energy, each pixel in an image is considered as a graph node and two nodes for foreground and background are added in the graph. Then, the data term is obtained by connecting each pixel to both the foreground and background nodes with non-negative edge weights represented by $D(x_i = 1)$ and $D(x_i = 0)$. The smoothness term is obtained by connecting each pairwise combination of neighboring pixels (i,j) with a non-negative edge weight represented by $V(x_i, x_j)$. We modified the graph-cut segmentation approach described in [11] by modifying the data term, $D(x_i)$ for tattoo segmentation. For skin detection, we define the data term as:

$$D(x_i) = w_1 D_1(x_i) + w_2 D_2(x_i), \tag{2}$$

where $D_1(x_i)$ is the energy for the skin or non-skin color model, $D_2(x_i)$ is the energy for the visual saliency map, $x_i = 1$ means skin, $x_i = 0$ means non-skin, and w_1 and w_2 are weights for $D_1(x_i)$ and $D_2(x_i)$. Then, each energy term is defined as:

$$\begin{aligned} D_1(x_i = 1) = -log(p_1(x_i = 1)), \quad D_2(x_i = 1) = -log(p_2(x_i = 1)). \quad for \quad i \in PS \\ D_1(x_i = 1) = \infty, \quad D_2(x_i = 1) = \infty. \quad for \quad i \in I - PS \end{aligned} \tag{3}$$

where PS are the possible segmentation regions.

Since the each mixture component of the GMM can model the different skin tone properly, we use a Gaussian Mixture Model (GMM) for $p_1(x_i = 1)$ to train the skin color model on diverse skin tones. It is defined as:

$$p(x_i = 1) = \sum_{j=1}^{M} \pi_j * g(C_i|u_j, \Sigma_j) \tag{4}$$

where C_i is the YCbCr color of i^{th} pixel, π_j is the j^{th} mixture weight, M is the number of mixture components, and $g(C_i|u_j, \Sigma_j)$, j=1,...,M, are the component Gaussian densities. The component Gaussian density, $g(C_i|u_j, \Sigma_j)$ is defined as:

$$g(C_i|u_j, \Sigma_j) = \frac{1}{(2\pi)^{3/2}|\Sigma_j|^{1/2}} exp\{\frac{1}{2}(C_i - u_j)^T \Sigma_j^{-1}(C_i - u_j)\} \tag{5}$$

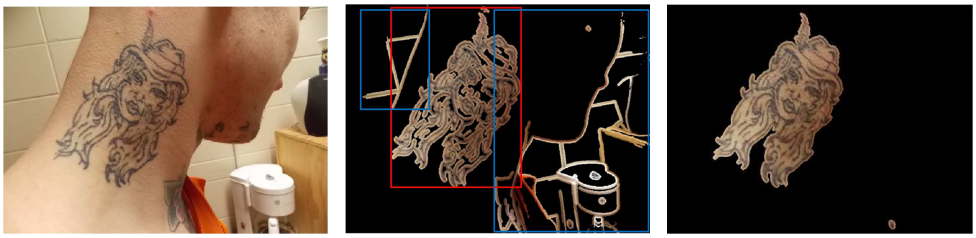
To estimate the skin color model parameters, u_j, Σ_j, and π_j we use EM (Expectation-Maximization) [12] on a skin image dataset obtained from [13]. For $p_2(x_i = 1)$, we use the visual saliency map from the GBVS (Graph-based visual saliency) [14]. To generate the saliency map from the GBVS feature vectors are extracted using the method of [15] first. Low-level visual features(color, contrast, and an orientation) are used as the feature vectors with a Gaussian pyramid in [15]. Then, it generates each saliency map(activation map) using each feature vector. Lastly, it normalizes each saliency map and combines all saliency maps together using graph-based approach. Since the GBVS saliency map does not have the same resolution as the image we upsample the map using bicubic interpolation. We do not use the GMM to train the non-skin color model because we cannot estimate all non-skin colors. Instead, we use an adaptive threshold value for $D_1(x_i = 0)$. Similarly, another adaptive threshold value is used for $D_2(x_i = 0)$:

$$\begin{aligned} D_1(x_i = 0) = -log(wt_1 * \frac{1}{n_I} \sum_{i \in I} p_1(x_i = 1)), \quad D_2(x_i = 0) = -log(wt_2 * \frac{1}{n_I} \sum_{i \in I} p_2(x_i = 1)), \ for\ i \in PS \\ D_1(x_i = 0) = 0, \qquad D_2(x_i = 0) = 0. \qquad for \qquad i \in I - PS \end{aligned} \tag{6}$$

where I is an image, n_I is the number of pixels in I, and wt_1 and wt_2 are weights for thresholds. Since we use adaptive thresholds instead of fixed thresholds for $D_1(x_i = 0)$ and $D_2(x_i = 0)$ the labeling errors can be reduced when the skin colors in an image do not fit our skin color training model or a GBVS does not fit the image. The smoothness term, $V(x_i, x_j)$ is defined as:

$$V(x_i, x_j) = |x_i - x_j| * f(C_{ij}) \tag{7}$$

where $f(\xi) = \frac{1}{1+\xi}$, and $C_{ij} = ||C_i - C_j||^2$ is the L2-Norm of the YCbCr color difference of two pixels i and j. This smoothness term is similar to the smoothness term in [11], but we use YCbCr color space for C_i while the RGB color space was used for C_i in [11]. The overall Gibbs energy, $E(x)$ is then minimized by min-cut/max flow in [16]. The min-cut/max flow algorithm consists of three stage: growth stage, augmentation stage, and adoption stage. In the growth stage it expands the search trees S and T until they touch each other with generating an $s - t$ path(s : a node for foreground, t : a node for background). In augmentation stage the path found in the growth stage is augmented. The adoption stage restores single-tree structure of sets S and T with roots in s and t.



(a) Original Image (b) Graph-Cut Segmentation (c) Final Tattoo Segmentation

Figure 3: Examples of false contours: In 3b, the red rectangular box is the minimum bounding box for a tattoo region, and the blue rectangular boxes are the minimum bounding boxes which include false contours. Using additional constraint, (8), a tattoo region is only segmented as depicted in Figure 3c

2.3 Post-Processing

From the graph-cut segmentation the skin pixels near edges are detected as shown in Figure 2e. Our method also detects skin pixels near boundaries of the body as well as the skin pixels near tattoos. To extract the skin pixels only near tattoos we check which set of skin pixels connected with each other forms a closed contour with a hole inside because the tattoo pixels surrounded by skin pixels will be labeled as non-skin and it will formulate holes. We can find the skin pixels near tattoos by finding the connected components of segmented pixels which have holes inside. As shown in Figure 3b, however, skin pixels near boundaries of the body might formulate closed contours with small holes inside because of segmentation errors. We will call these closed contours false contours. To distinguish a closed contour near tattoos with the false contours we additionally check the ratio of the area of minimum bounding rectangular box including closed contours to the area surrounding all pixels connected with a closed contour.

$$\frac{n_c + n_h}{n_b} >= t_f \tag{8}$$

where n_c is the number of pixels in a closed contour, n_h is the number of pixels for a hole inside the closed contour, n_b is the number of pixels in the minimum bounding rectangular box including the closed contour, and t_f=0.35 is the threshold to detect a false contour. If a closed contour satisfies Equation (8) the closed contour with inside regions is considered to be a tattoo region.

3. EXPERIMENTAL RESULTS

To evaluate our segmentation method the tattoo images acquired from the Indiana Gang Network (INGangNetwork) are used. Some of these images are shown in Figure 5 and Figure 6. Since we do not have ground truth for tattoo segmentations we show that our proposed method can segment tattoo regions well in several images by showing several segmentation results. Also, we compared our method with [5]. In our experiments we also used the TDSD dataset [13] which includes diverse skin images to train the GMM for our skin color model. The mixture number of the GMM, M =5 for our experiments. Since the purpose of the tattoo segmentation is to extract tattoo regions from a tattoo image for an image retrieval system, our method does not segment the tattoo inside the closed contour we found after post processing. The tattoo segmentation results of our proposed method and [5] are shown in Figure 5 and Figure 6. As depicted in Figure 5 and Figure 6, our proposed method segmented most of tattoo images correctly while [5] made segmentation errors in several tattoo images. Even though images used in our experiments have diverse skin tones, our method correctly segmented tattoo regions from the images. When some images have skin-colored background with strong edges, our method can still segment tattoo region correctly. However, some segmentation results showed the errors depicted in Figure 4. When tattoo regions are very close to boundaries of the body, our method found the skin pixels connected to the tattoo regions and the boundaries of the body together, so our post processing could not extract a tattoo region only. Also, if there are hair regions with strong edges inside a body, our method also detected the hair regions as tattoo regions. We will address these problems in future work.

Figure 4: Incorrect Tattoo Image Segmentation. The first and fourth column images are original tattoo images, the second and fifth column images are segmentation results of [5], and the third and sixth column images are our proposed segmentation results

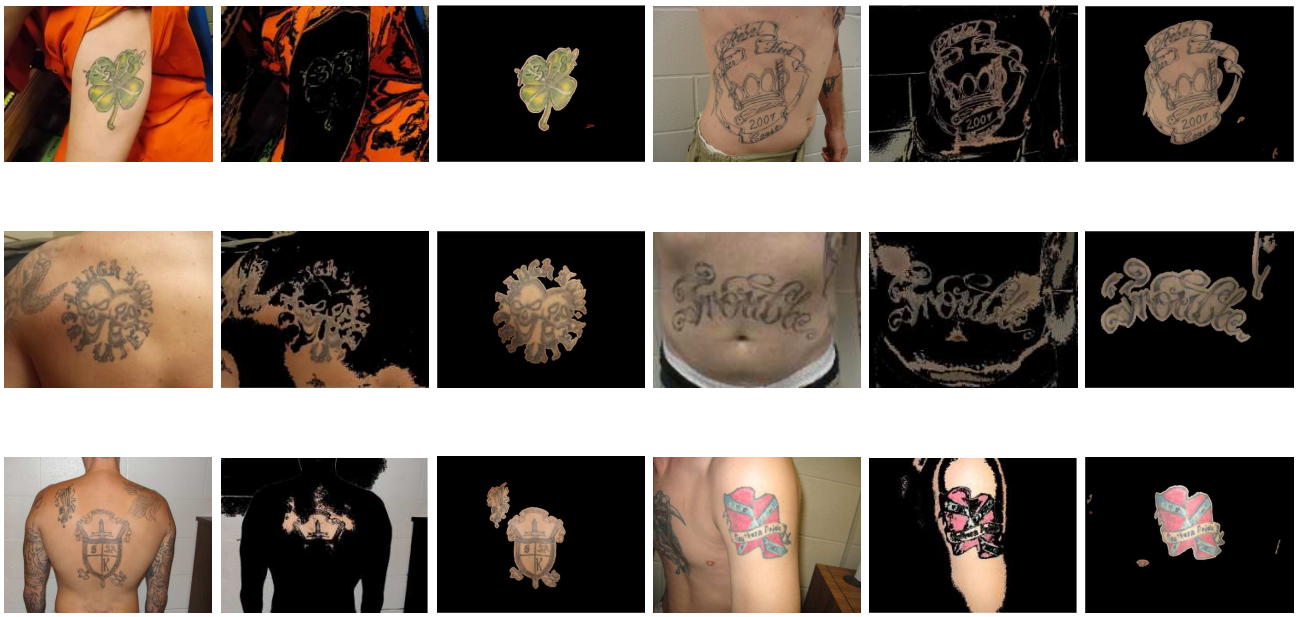


Figure 5: Tattoo Image Segmentation Results. The first and fourth column images are original tattoo images, the second and fifth column images are segmentation results of [5], and the third and sixth column images are our proposed segmentation results

4. CONCLUSIONS AND FUTURE WORK

In this paper we described a new tattoo segmentation approach by determining skin pixels around a tattoo. Only regions near image edges are considered as possible segmentation regions. In these regions graph-cut segmentation using a skin color model and a visual saliency map was used to find skin pixels. After the graph-cut segmentation we determine which set of skin pixels are connected with each other that form a closed contour including a tattoo. To remove false closed contours caused by graph-cut segmentation errors we additionally check the ratio of the area of minimum bounding rectangular box including closed contours to the area surrounding all pixels connected with a closed contour. The regions surrounded by the final closed contours are considered tattoo regions. In the experimental results we showed our method achieved more accurate tattoo segmentation compared to exiting

methods [5]. However, our method sometimes fails to extract tattoo regions correctly when there are tattoo regions very close to boundaries of the body and there are hair regions with strong edges inside a body. To address these problem we will investigate techniques to detect the boundaries of a body and the hair regions inside the body.

REFERENCES

[1] P. Duangphasuk and W. Kurutach, "Tattoo skin detection and segmentation using image negative method," *Proceedings of the International Symposium on Communications and Information Technologies*, pp. 354–359, September 2013, Surat Thani, Thailand.

[2] B. Heflin, W. Scheirer, and T.E. Boult, "Detecting and classifying scars, marks, and tattoos found in the wild," *Proceedings of the IEEE International Conference on Biometrics: Theory, Applications and Systems*, pp. 31–38, September 2012, Arlington, VA.

[3] D. Manger, "Large-scale tattoo image retrieval," *Proceedings of the Conference on Computer and Robot Vision*, pp. 454–459, May 2012, Toronto, ON.

[4] J. Lee, R. Jin, A.K. Jain, and W. Tong, "Image retrieval in forensics: Tattoo image database application," *IEEE Transactions on Multimedia*, vol. 19, no. 1, pp. 40–49, January 2012.

[5] J. Allen, N. Zhao, J. Yuan, and X. Liu, "Unsupervised tattoo segmentation combining bottom-up and top-down cues," *Proceedings of the SPIE Mobile Multimedia/Image Processing, Security, and Applications*, vol. 8063, pp. 1–9, April 2011, Orlando, FL.

[6] A.K. Jain, J. Lee, R. Jin, and N. Gregg, "Content-based image retrieval: An application to tattoo images," *Proceedings of the IEEE International Conference on Image Processing*, pp. 2745–2748, November 2009, Cairo, Egypt.

[7] J. Lee, R. Jin, and A.K. Jain, "Rank-based distance metric learning: An application to image retrieval," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2008, Anchorage, AK.

[8] S.T. Acton and A. Rossi, "Matching and retrieval of tattoo images: Active contour cbir and glocal image features," *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 21–24, March 2008, Santa Fe, NM.

[9] A.K. Jain, J. Lee, and R. Jin, "Tattoo-ID: Automatic tattoo image retrieval for suspect and victim identification," *Proceedings of the Pacific Rim Conference on Multimedia*, pp. 256–265, December 2007, Hong Kong, China.

[10] C. Rother, V. Kolmogorov, and A. Blake, "GrabCut: Interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics.*, vol. 23, no. 3, pp. 309–314, August 2004.

[11] Y. Li, J. Sun, C. Tang, and H. Shum, "Lazy snapping," *Proceeding of ACM SIGGRAPH*, pp. 303–308, August 2004, New York, NY.

[12] M. R. Gupta and Y. Chen, "Theory and Use of the EM Algorithm," *Foundations and Trends in Signal Processing*, vol. 4, no. 3, pp. 223–296, 2010.

[13] Q. Zhu, C. Wu, K. Cheng, and Y. Wu, "A unified adaptive approach to accurate skin detection," *Proceedings of the IEEE International Conference on Image Processing*, pp. 1189–1192, October 2004, Singapore, Singapore.

[14] J. Harel, C. Koch, and P Perona, "Graph-based visual saliency," *Proceedings of the Annual Conference on Neural Information Processing Systems*, pp. 545–552, December 2006, Vancouver, B.C., Canada.

[15] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, November 1998.

[16] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, September 2004.

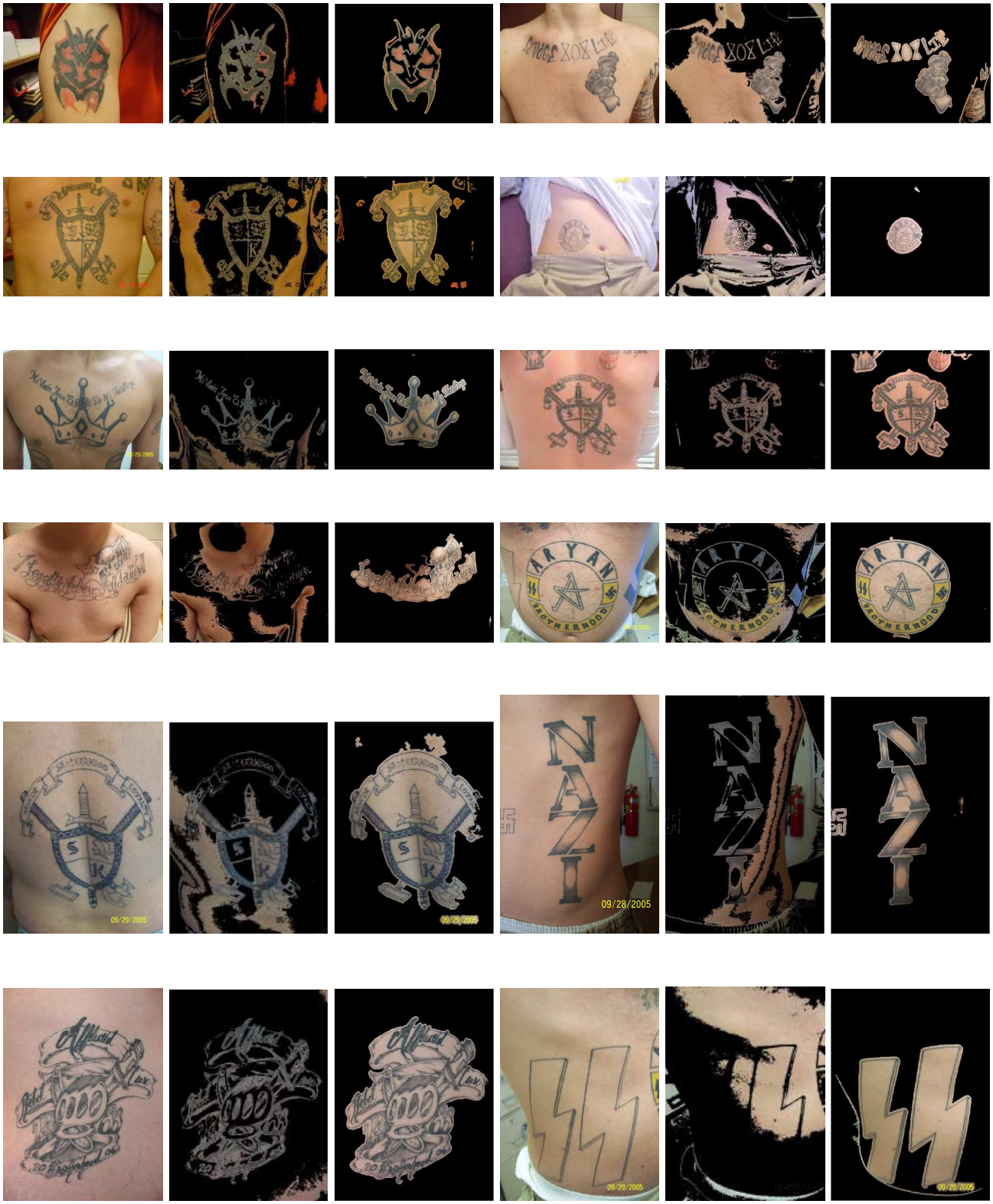


Figure 6: Tattoo Image Segmentation Results. The first and fourth column images are original tattoo images, the second and fifth column images are segmentation results of [5], and the third and sixth column images are our proposed segmentation results

approach to hazmat sign location detection using a new visual saliency model proposed in Chapter 2 and a Fourier descriptor based contour matching method [74]. We use the best of our proposed frequency domain models to extract salient regions that are likely to contain hazmat sign candidates and then use a Fourier descriptor based contour matching method to locate the border of hazmat signs in these regions. This visual saliency based approach is able to significantly increase the accuracy of sign location detection, reduce the number of false positive objects, and speed up the overall image analysis process. It uses contour-based shape representation and correlation matching based on the magnitude and phase of the Fourier descriptor of extracted contours. Closed contours are extracted from color channel images using adaptive thresholding, image binarization, morphological operation and connected component analysis. Experimental results show that our proposed hazmat sign location detection method is capable of detecting and recognizing projective distorted, blurred, and shaded hazmat signs in complex scenes.

3.1 Review of Existing Sign Detection and Recognition Methods

3.1.1 Sign Location Detection

Sign location detection approaches can be divided into three categories: color-based methods [75], shape-based methods [76] and vision-based methods [24]. Color-based methods take advantage of the fact that signs often have highly visible contrasting colors. These specific colors are used for sign location detection. For example, a color histogram backprojection method is used in [77] to detect interesting regions possibly containing hazmat signs. In [78] sign location detection is performed using a color-based segmentation method as a preprocessing step for shape detection. The luminance homogeneity of blocks is used in [79] to identify homogenous regions as the first step towards detection of information signs containing text. In [80] several color components are used to segment traffic signs in various weather conditions. However, color-based methods are not robust to lighting conditions and illumination changes.

Shape-based approaches first generate an edge map and then use shape characteristics to find signs. For example, in [81] triangular, square and octagonal road signs are detected by exploiting the properties of symmetry and edge orientations exhibited by equiangular polygons. A shape classification method of a road-sign detection system in [82] is based on linear and Gaussian-kernel support vector machines (SVM). In [83] the authors present a system for detection and recognition of road signs with red boundaries and black symbols inside. Pictograms are extracted from the black regions and then matched against templates in a database. They propose a fuzzy shape detector and a recognition approach that uses template matching to recognize rotated and affine transformed road signs. In [84] the authors propose a system for automatic detection and recognition of traffic signs based on maximally stable extremal regions (MSERs) and a cascade of SVM classifiers trained using histogram of oriented gradient (HOG) features. The system works on images taken from vehicles, operates under a range of weather conditions, runs at an average speed of 20 frames per second, and recognizes all classes of ideogram-based (nontext) traffic symbols from an online road sign database. In most cases shape-based methods are invariant to translation, rotation, and scaling, while in some situations to partial occlusions. Because color-based or shape-based sign location detection methods have both strengths and disadvantages, most color-based approaches take shape into account after using color features while some shape-based detectors also integrate some color aspects.

Vision-based approaches approaches utilize selective visual attention models, which imitate human early visual processing in order to overcome the above problems in complex scenes. Many vision-based traffic sign location detection and analysis methods are using visual saliency models to generate saliency maps that denote areas where signs are likely to be found [24]. For example, in [85] a saliency map of road traffic signs is constructed by a weighted sum of color and edge feature maps. A traffic sign recognition system in [86] uses a visual attention system to denote regions with possible candidates. In our previous work [14, 15] we proposed several image

analysis methods using visual saliency for hazmat sign location detection and context recognition. This extended work makes use of our proposed visual saliency models to construct a saliency map as a part of hazmat sign location detection method.

3.1.2 Sign Recognition

Sign recognition methods can be classified into: geometric constraint methods, boosted cascades of features, and statistical moments [24, 79, 87].

Methods based on geometric constraints include the use of Hough-like methods [88, 89], contour fitting [90, 91], or radial symmetry detectors [92, 93]. These approaches apply constraints on the object to be detected, such as little or no affine transformations, uniform contours, or uniform lightning conditions. Although these conditions are usually met, they cannot be generalized. For example, [89] presents an analysis of Hough-like methods and confirms that the detection of signs under real-world conditions is still unstable. A novel Hough-like technique for detecting circular and triangular shapes is also proposed, in order to overcome some of the limitations exposed.

Methods based on the boosted cascades of features commonly use the Viola-Jones framework [94–96]. These approaches often use object detectors with Haar-like wavelets of different shapes, and produce better results when the feature set is large. For example, in [95] a system for detection, tracking, and classification of U.S. speed signs is presented. A classifier similar to the Viola-Jones detector is used to discard objects other than speed signs in a dataset of more than 100,000 images. In [96] the detection is based on a boosted detectors cascade, trained with a version of Adaboost, which allows the use of large feature spaces. The system is robust to noise, affine deformation, partial occlusions, and reduced illumination.

Methods based on statistical moments [97–99] use the central moments of the projections of the object to be detected. They can be used to check the orientation of the object, or to distinguish between different shapes such as circles, squares, triangles,

or octagons. These methods are not robust to projective distortions or non-uniform lightning conditions. For example, in [99] a mobile-based sign interpretation system uses detection of shapes with an approximate rotational symmetry, such as squares or equilateral triangles. It is based on comparing the magnitude of the coefficients of the Fourier series of the centralized moments of the Radon transform of the image after segmentation. The experimental results show that the method is not robust to projective distortions.

3.1.3 Shape Descriptors

Shape is an important low level object and image feature [76, 100–102]. Shape can be described using "shape descriptor," which can be generally classified into two methods: contour-based methods and region-based methods [103]. Contour-based methods only exploit the boundary information while region-based methods exploit all the pixels within a region. Contour-based methods are widely used in many applications because of their computational efficiency but they may fail when objects have low resolution. The Fourier descriptors (FD) is a classic and still popular method for contour description [104, 105]. The key idea is to use the Fourier transform of the periodic representation of the contour, which results in a shape descriptor in the frequency domain. The low-frequency components of the descriptor contain information about the general shape of the contour while the finer details are described in the high-frequency components [74]. Although shape descriptors obtained from contour-based methods are not generally robust to noise [106], the Fourier descriptor overcomes noise sensitivity by usually using only the first few low frequency coefficients to describe the shape [74, 103, 107]. The FD is also compact and easy to normalize. In addition, it has been shown that the FD outperforms many other shape descriptors [106, 108].

Existing work on Fourier descriptor (FD) includes methods for generating descriptors invariant to geometric transformations and matching methods for shape similarity and matching. For example, a new Fourier descriptor for image retrieval

is proposed in [109] by exploiting the benefits of both the wavelet and Fourier transforms. A complex wavelet transform is first used on the shape boundary and then the Fourier transform of the wavelet coefficients at multiple scales is employed. Since FD is used at multiple scales, the shape retrieval accuracy improves with respect to using ordinary FD. FD feature vectors are analyzed for pedestrian shape representation and recognition [110]. The results show that only ten descriptors of both low and high frequency components of pedestrian and vehicle shapes are enough for accurate shape recognition. The fast FD of some shapes are presented in [111] based on chain codes and the Fourier transform for shape recognition. It is shown that the first ten terms of Fourier coefficients are enough to approximate the shapes. In [74] a method using the Fourier transform of local regions is developed to describe the contours in these regions. A correlation-based contour matching method is also proposed in [74] using both magnitude and phase information of Fourier descriptors for recognizing road signs.

3.2 Review of Existing Hazmat Sign Detection and Recognition Systems

Although there exist several mobile-based applications that provide easy access to the Emergency Response Guidebook (ERG) guidebook [1, 12], they only provide manually browsing functionality. There are a few methods in the literature dealing with sign detection and recognition, but we are only aware of two other published papers with application to hazmat signs [77, 112].

3.2.1 Hazmat Sign Detection Based on SURF and HBP

In [77] the hazmat sign detection is done using color histogram back-projection (HBP) and Speeded Up Robust Feature (SURF) [113] matching. The method was implemented and tested on an autonomous mobile robot for the 2008 RoboCup World Championship. Histogram back-projection is used to detect regions of interest in the image and remove the background of the scene. A background image without a sign,

$h(x, y)$, is used as a ground-truth to isolate the hazmat sign when it appears on the scene and an image of it is captured, $f(x, y)$. This is done by determining the euclidean distance of the color coordinates of each pixel within $h(x, y)$ and the corresponding pixel within $f(x, y)$. A threshold K is used to create a binary mask of the hazmat sign by the use of an indicator function $\delta(x, y) = \{(x, y)\ s.t.\ |f(x, y) - h(x, y)| > K\}$. Several color histograms are then estimated for the U and V channels on the YUV color space, and summed up to create a single histogram $H_o(U, V)$ for every sign on the image. A threshold $\theta(H_o, \epsilon)$ is used for $H_o(U, V)$, resulting in a binary indicator function $\pi_o(U, V)$, which specifies which pixels form part of a sign. The value of ϵ is manually set to 0.05. Finally, morphological filters are used to segment the masked regions from the background and create one or more regions of interest to be used as inputs to the matching process using SURF features.

SURF matching is used to find interest points and retrieve images from a database. After the region of interest is determined from the image containing a hazmat signs, multiple interest points are found using SURF. Interest points surrounding regions that overlap the region of interest are discarded, since the do not provide enough information about the sign. For the remaining interest points, their corresponding feature vectors are matched against all features of all images in a database corresponding to the colors found on the first step.

The experiments were done using a stereo camera system consisting of two cameras with a resolution of 1024×768 pixels. The tests consisted of detecting five different hazmat signs in 240 images. The images were taken at 1, 1.5 and 2 meters, with a maximum distortion of $30°$. The results show a detection accuracy of 92% from 1 meter, 52% from 1.5 meters, and less than 20% from 2 meters. The running time ranges from 1 to 1.6 second on a 2.7GHz Intel CPU.

3.2.2 Hazmat Sign Detection Based on HOG

In [112] hazmat sign detection using sliding windows and Histogram of Oriented Gradients (HOG) [114] is described. The method was implemented and tested on a wheeled USAR robot for the 2010 RoboCup World Championship.

The authors use the sliding window approach to exhaustively scan every pixel over a range of positions and scales, with steps of 8 pixels and relative scale factors of 1.05. For each position and scale a discriminative Support Vector Machine (SVM) classifier is used to make binary decisions about the presence or absence of an object. In order to describe the contents of the image at each particular location a HOG descriptor is used along with color histograms in the Lab color space to distinguish between multiple hazmat signs. For each hazmat sign hypothesis of the HOG based detector, the color histogram is used to do the final classification by applying a k-nearest neighbor approach in combination with χ^2-distance.

The experimental results show a recognition rate of 37.5% using histograms based on entire sliding windows and a recognition rate of 58.3% using sub-region based histograms. Region-based histograms provide better representation of the image since they are capable of capturing the spatial distribution of colors within the detection window.

3.2.3 Comparison to MERGE

We proposed a hazmat sign location detection and content recognition system, known as MERGE (Mobile Emergency Response GuidE) [13]. Although all methods above are deployed on mobile environments, MERGE is intended for real-time use by first responders, while [77] and [112] were intended for use in a very specific context. The sign detection method proposed in [77] uses a ground-truth image of the background to aid in detection when the hazmat sign appears. This is not a feasible assumption in MERGE, since the first responders are expected to take images of hazmat signs in a large variety of scenarios. In [112] a dataset of 1480 daylight

images is used for both people and hazmat sign recognition. However, the authors do not specify how many images contain hazmat signs, or at what distances the signs are located. They do not provide information about the resolution of the images or the cameras used for acquisition. In MERGE no assumptions on the background are made in order to detect the sign. Instead, color information is used to detect candidate regions using a saliency map model.

Once the hazmat sign is detected [77] uses image matching based on SURF features, and [112] uses HOG and color histogram descriptors, both being very time consuming task. This step is not done in MERGE. Currently, the color of the hazmat sign is considered to be uniform, and the detection is made at different color channels. The recognition of non-uniformly-colored placards is presented as part of the future work.

The goal of MERGE is to be able to detect hazmat signs at long distances. Our experimental results show successfully detecting hazmat signs in some cases at more than 100 feet. However, the experiments in [77] can only be considered successful at 1.5 meters, and the accuracy reported by [112] is very low. Finally, the execution time of the overall process of our hazmat sign image analysis system MERGE is several seconds, comparable with the hazmat sign image analysis system in [77]. No execution time is reported in [112].

3.3 Proposed Hazmat Sign Detection and Recognition System

3.3.1 MERGE System Overview[1]

Figure 3.2 shows the overview of our proposed hazmat sign location detection and content recognition system, known as MERGE (Mobile Emergency Response GuidE) [13]. It consists of an application running on an Android/iOS mobile device[2] and a backend server where many image analysis operations are done [14, 15]. There are two basic operational modes of our MERGE system: analysis of hazmat sign images and searching internal database. The first mode includes capturing or selecting an hazmat sign image from the mobile device and performing image analysis on the backend server. Hazmat sign detection and recognition are done on the backend server and the results are sent back to the mobile device [14, 15]. The second mode includes searching the internal database to obtain guide information about a specific hazmat sign. We designed an internal database based on the contents of the 2012 ERG guidebook. As shown in Figure 3.3, hazmat signs can be manually searched by UN identifier numbers, template images, symbols, and classes.

Figure 3.4 shows the operational workflow and user interface at each step. The image analysis results are used for matching related guide pages and querying internal database to retrieve guide information. We display guide information about potential hazards, public safety and emergency response. All the information is from the internal database on the mobile application. A suggested evacuation region is also displayed on a map based on the chemical found, the size of the chemical spill, the time of the day, and a weather-aware chemical spreading webservice.

[1]The work in this section was developed by the author jointly with my colleagues Albert Parra and Joonsoo Kim.

[2]The Android application was developed by my colleague Albert Parra and the iOS application was developed by my colleague Joonsoo Kim.

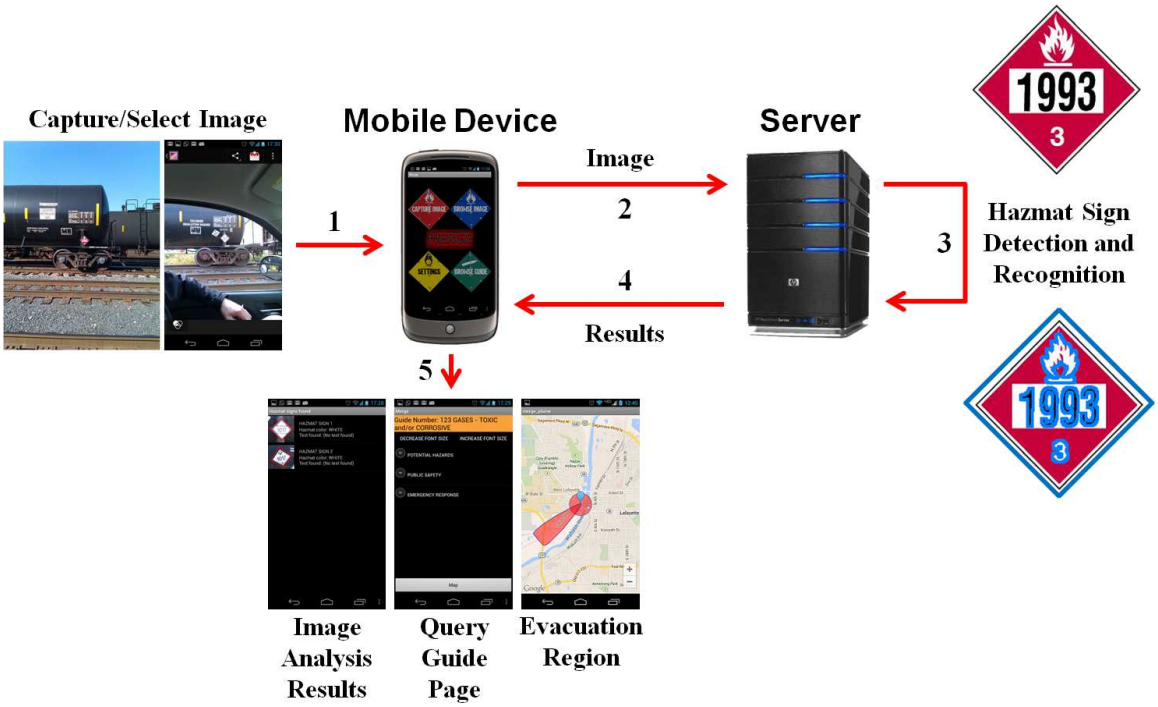


Fig. 3.2. Hazmat sign location detection and content recognition system.

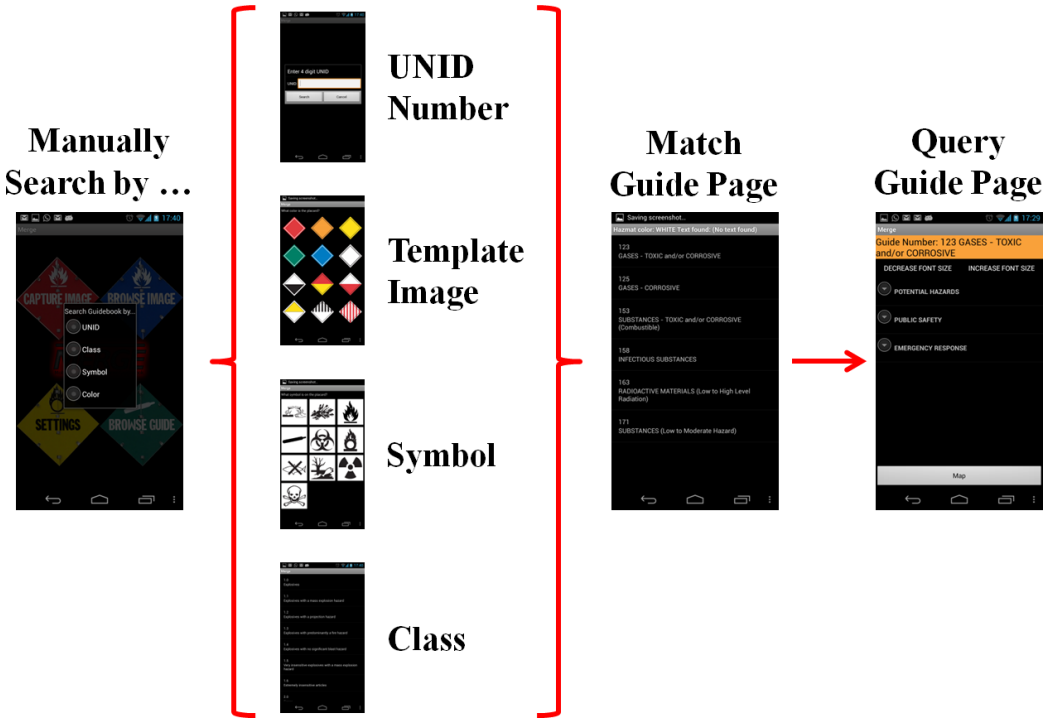


Fig. 3.3. Manually search hazmat signs by UN identifier numbers, template images, symbols, and classes.

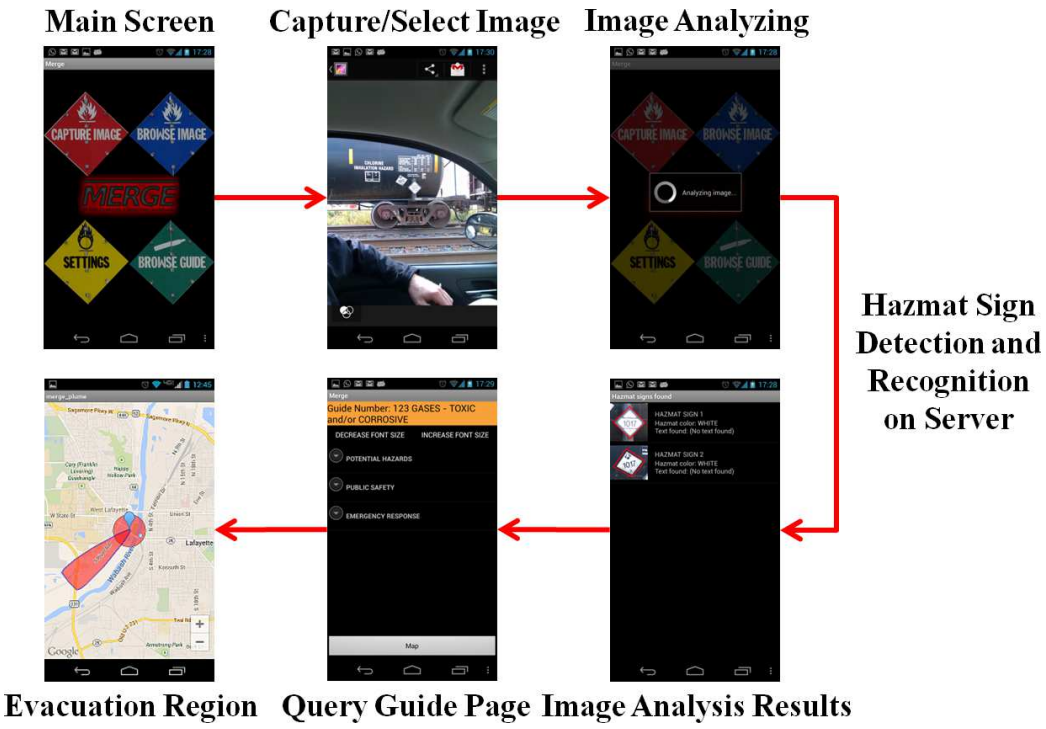


Fig. 3.4. Mobile application user interface at each step.

3.4 Hazmat Sign Detection and Recognition Method 1

We use visual saliency based methods and generate saliency maps using color spaces. Spatial domain visual saliency models usually have high computational cost and variant parameters for multiple feature maps, which make them impractical to meet our needs. Frequency domain visual saliency models with fast computation with high prediction accuracy could be suitable for our application. Our proposed hazmat sign detection and recognition method is based on visual saliency. We use two existing visual saliency models to generate saliency maps denoting salient regions likely containing hazmat signs in complex scenes and develop a convex quadrilateral shape detection method to extract the border of hazmat signs in these regions. The block diagram in Figure 3.5 shows the building blocks of the proposed hazmat sign detection and recognition method 1.

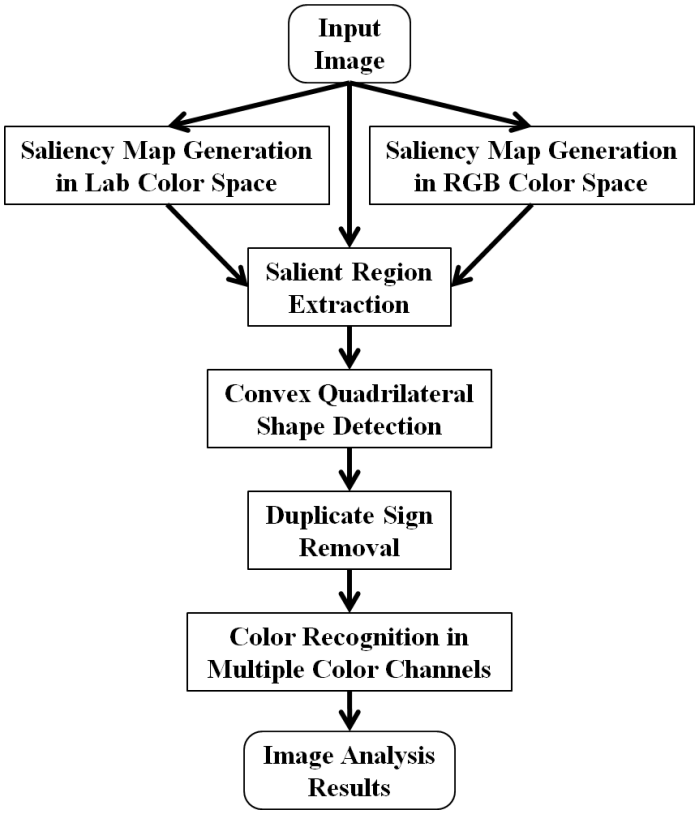


Fig. 3.5. Proposed hazmat sign detection and recognition method 1.

3.4.1 Saliency Map Generation

We use two existing visual saliency models to generate saliency maps from images represented in both Lab and RGB color spaces, because we observed that color signs have strong visual responses in Lab color space while white signs have strong visual responses in RGB color space from our experiments. In each color space, two saliency maps are generated separately using two visual saliency models, *i.e.* IS model [33] and SSA model [35] respectively. The saliency maps assign higher saliency value (ranging from lowest 0 to highest 1) to more visually attractive regions that are likely containing hazmat signs in complex scenes. Note that the original SSA method uses the IRGBY color space [35]. We modified this method to use Lab and RGB color components with different weights ($[\frac{1}{2}, \frac{1}{4}, \frac{1}{4}]$ for Lab and $[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$ for RGB). The proposed hazmat sign detection method using the four saliency maps (two from Lab and two from RGB), denoted as the combined method IS+SSA(Lab+RGB), has good performance in the experiments. (see Section 3.6)

3.4.2 Salient Region Extraction

We threshold each saliency map to create a binary mask to extract the salient regions from the original image. The threshold T_1 is determined as k times the average saliency value of a given saliency map.

$$T_1 = \frac{k}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} S(x, y), \tag{3.1}$$

where W and H are the width and height of the saliency map, $S(x, y)$ is the saliency value at position (x, y) and k is empirically determined for the combined detection method IS+SSA(Lab+RGB), *i.e.* $k = 4.5$ for IS models and $k = 3.5$ for SSA models, which provides a good trade-off between hazmat sign coverage and computational cost of extracted salient regions in the experiments. (see Section 3.6) The

following processing can take advantages of local distinctive features in the extracted salient regions instead of the entire input image.

3.4.3 Convex Quadrilateral Shape Detection[3]

For each salient region found, we detect hazmat sign candidates in specific color channels. We used black and white information from grayscale image, and red, green and blue channels from RGB color space. Note that the possible colors for hazmat signs also include yellow and orange, but these can be obtained by transforming the image from RGB to a hue-based color space and then segment the hue channel. The grayscale image and the color channels are thresholded to account for highly chromatic areas using an empirically determined threshold T_2 (85 for black, 170 for white, and 127 for color). Each binarized region is morphologically opened to remove small objects and morphologically dilated to merge areas that may belong to the same object. We then retrieve contours from the resulting binary image using the border following technique proposed in [115]. For each contour, we use the Hough Transform [116] to find straight lines that approximate the contour as a polygon. The intersections of these lines are the corners of the polygon which can be used to discard non-quadrilateral shapes. If the contour is approximated by four vertices, we find its convex hull [117]. If the convex hull still has four vertices, we check the angles formed by the intersection of its points. If each of these angles is in the range $90° \pm 1.5°$, and the ratio of the sides formed by the convex hull is in the range 1 ± 0.5, we assume that the convex quadrilateral is a hazmat sign candidate.

3.4.4 Duplicate Sign Removal

To remove duplicate sign candidates from different color channel images, we first check all candidates passed the contour matching and estimate their minimal bounding boxes. Any disqualified candidate with the aspect ratio of its bounding box

[3]The work in this section was developed by the author jointly with my colleague Albert Parra.

greater than 1.3 will be discarded. We then remove the duplicate sign candidates that correspond to the same sign. This can be done by first dividing all candidates that are overlapped more than 50% into multiple groups and then finding the optimal diamond-shaped box for each group, whose four nodes are closest to the centroid of its group. Each optimal diamond-shaped box is considered to be the location of a detected hazmat sign.

3.4.5 Color Recognition[4]

Because signs are detected in specific color channels, the color is recognized directly from the color channel where the sign was identified (black or white for grayscale and red, green or blue for RGB). The recognized color is used for queuing the mobile database for sign category identification and providing the general guide information based on the 2012 ERG guidebook. Figure 3.6(a) illustrates a successful detection of two signs using the previous Method 1, one of which is affected by projective and rotational distortion. Figure 3.6(b) illustrates a true positive and a false positive from the previous Method 1.

(a) Two true positives. (b) One true positive and one false positive.

Fig. 3.6. Examples of image analysis.

[4]The work in this section was developed by the author jointly with my colleague Albert Parra.

3.5 Hazmat Sign Detection and Recognition Method 2

We use our proposed frequency domain models in Chapter 2 to extract salient regions that are likely to contain hazmat sign candidates and develop a Fourier descriptor based contour matching method to extract the border of hazmat signs in these regions. Based on our previous work [14], we propose a new approach to hazmat sign location detection using a Fourier descriptor based contour matching method [74]. It uses contour-based shape representation and correlation matching based on the magnitude and phase of the Fourier descriptor of extracted contours. The existing method used to detect road signs in [74] cannot be directly used for hazmat sign location detection. Hazmat signs mounted on vehicles are usually enclosed in a placard holders with two horizontal strips that divide a hazmat sign into three separate parts as shown in Figure 3.1. In our case we need to use morphological operations to merge separate parts that belong to a whole hazmat sign and then employ connected component analysis to determine the boundary of the whole hazmat sign. Closed contours are extracted from color channel images using adaptive thresholding, image binarization, morphological operation and connected component analysis.

Fourier Descriptor (FD) is used to describe the shape of the extracted contours through the Fourier transform [104, 105]. It has been proven to be a state-of-the-art contour-based sign detection methods in terms of accuracy and tolerance of rotated, scaled, and noisy signs [74, 105, 110]. In order to determine if an extracted contour correspond to a hazmat sign, we need to compare its FD against the FD of the contour of a shape template or a predefined shape contour. In our case, the shape template of hazmat signs is represented by a diamond shaped binary image as shown in Figure 3.7.

Contour matching can be done in the spatial or frequency domain. We use matching in the frequency domain for two reasons. First, matching in the frequency domain is scale independent, as opposed to spatial domain matching. Second, matching in the spatial domain involves scanning an image multiple times modifying the scale

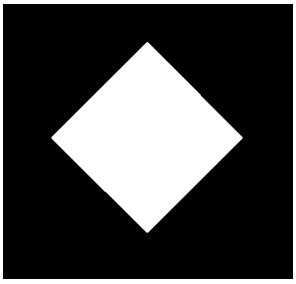


Fig. 3.7. A diamond shaped binary image represents the shape template of a hazmat sign.

and rotation of the shape template. Frequency domain matching methods have been shown to be more computationally efficient when working with images of high resolution [118, 119]. FD-based matching is usually done by using only the magnitude and ignoring the phase information. By discarding the phase information, rotation and starting point invariance can be achieved [120]. However, because variant shapes can have similar magnitude but different phase information, this makes FD-based magnitude-only matching less accurate [74]. A correlation-based contour matching method is proposed in [74] using both magnitude and phase information of Fourier descriptors for recognizing road signs. It is shown that the normalized FDs are invariant to scaling and the correlation-based contour matching using both magnitude and phase information is invariant to rotation and starting point. We use this frequency domain contour matching method [74] to detect the location of hazmat signs based on a diamond shaped template. The block diagram in Figure 3.8 shows the building blocks of the proposed hazmat sign detection and recognition method 2.

3.5.1 Saliency Map Generation

We use our proposed frequency domain models to generate saliency maps from input images represented in both Lab and RGB color spaces, because we observed that color signs have strong visual responses in Lab color space while white signs have strong visual responses in RGB color space from our experiments. In each color

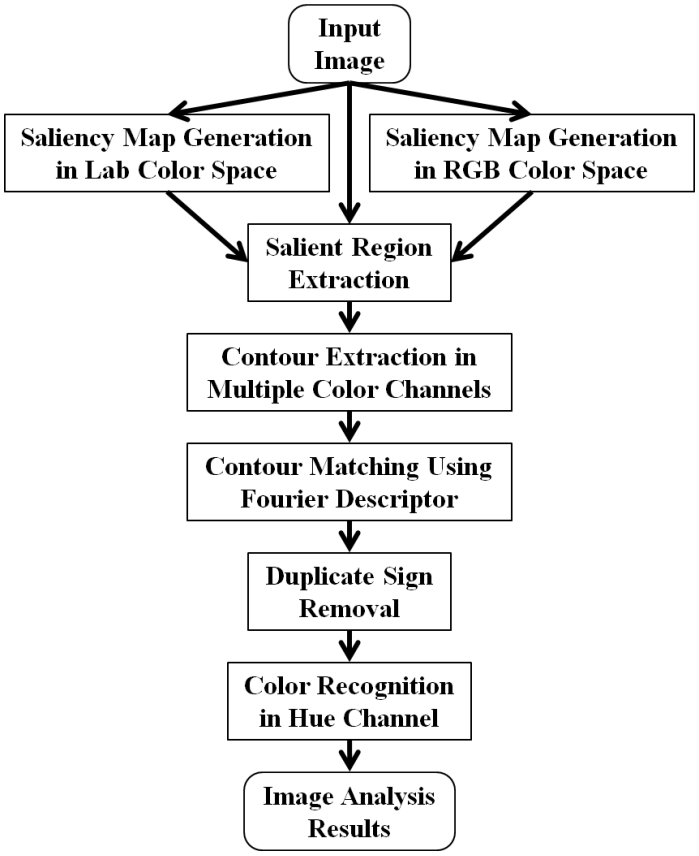


Fig. 3.8. Proposed hazmat sign detection and recognition method 2.

space, a saliency map is generated separately using the proposed Gamma Corrected Spectrum (GCS) visual saliency model, *i.e.* either GCS-FT-Lab model or GCS-FT-RGB model. (see Section 2.3.4) Two saliency maps, one from Lab color space and the other from RGB color space, assign higher saliency value (ranging from lowest 0 to highest 1) to more visually attractive regions that are likely containing hazmat signs in complex scenes. The proposed hazmat sign detection method using the two saliency maps, denoted as the combined method GCS(Lab+RGB), has the best performance in our experiments. (see Section 3.6)

3.5.2 Salient Region Extraction

We threshold each saliency map to create a binary mask to extract the salient regions from the original image. The threshold T_1 is determined as k times the average saliency value of a given saliency map.

$$T_1 = \frac{k}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} S(x, y), \tag{3.2}$$

where W and H are the width and height of the saliency map, $S(x, y)$ is the saliency value at position (x, y) and k is empirically determined for the combined detection method GCS(Lab+RGB), *i.e.* $k = 2.0$ for GCS-FT-Lab model and $k = 2.0$ for GCS-FT-RGB model, which provides a good trade-off between hazmat sign coverage and computational cost in extracted salient regions in our experiments. (see Section 3.6) The following processing can take advantages of local distinctive features in the extracted salient regions instead of the entire input image.

3.5.3 Contour Extraction

The hazmat signs in our dataset contain either one or two of the following colors: white, red, green, blue, and yellow. In order to obtain strong visual responses of certain colors of hazmat signs, we transform an extracted salient region of the input image into several channel images in different color spaces. The white signs can be detected in the grayscale channel image. The red, green and blue signs can be detected in R, G, B channel images from the RGB color space. The yellow signs can be detected in the Y channel image from the CMYK color space. We process each channel image of the extracted salient region separately in the following.

In order to binarize each channel image of the salient region A_k^{SR}, we propose a new adaptive thresholding method that is a modification of Otsu's thresholding method [121]. Since images containing hazmat signs are likely acquired with various lighting conditions, directly using Otsu's thresholding method on the channel image

does not produce accurate results when images contain variable illumination [122]. For each channel image, I_i, $i \in [1,5]$, we first use a histogram of 256 bins for the [0,255] grayscale values to characterize pixel distribution and then obtain the median of the pixel counts of all bins N_i^{MED}. Second, we find the starting location of two significant peaks T_i^L and T_i^H at the low and high ends of the histogram by checking the change of pixel counts between two adjoining bins. The two index thresholds T_i^L and T_i^H are selected to clip the histogram.

$$N_i^{MED} = median\left(\mathcal{N}(B_i^j)\right), \tag{3.3}$$

$$T_i^L = argmin_j\left(|\mathcal{N}(B_i^j) - \mathcal{N}(B_i^j - 1)| > F_B \cdot N_i^{MED}\right), j \in [3,128], \tag{3.4}$$

$$T_i^H = argmax_j\left(|\mathcal{N}(B_i^j) - \mathcal{N}(B_i^j + 1)| > F_B \cdot N_i^{MED}\right), j \in [129,254], \tag{3.5}$$

where $\mathcal{N}(B_i^j)$ is the pixel count of the j-th bin in the histogram of the i-th channel image, $F_B = 0.05$ is a factor to determine index thresholds T_i^L and T_i^H with respect to N_i^{MED} (empirically obtained by searching good values in our experiments), T_i^L is the starting location of the low-end significant peak (the index threshold of a low-end bin), and T_i^H is the starting location of the high-end significant peak (the index threshold of a high-end bin). For each color channel image, we modify its histogram by clipping the pixel counts $\mathcal{N}(B_i^j)$ of the low-end and high-end bins into 0s based on the two index thresholds T_i^L and T_i^H.

$$\mathcal{N}'(B_i^j) = \begin{cases} 0 & B_i^j \leq T_i^L \text{ or } B_i^j \geq T_i^H \\ \mathcal{N}(B_i^j) & \text{otherwise} \end{cases} \tag{3.6}$$

The modified histogram with new pixel counts $\mathcal{N}'(B_i^j)$ for all 256 bins is used with the original Otsu's method [121] to generate an adaptive threshold T_i^{BW}. Finally, each original channel image I_i is then binarized using T_i^{BW}. Figure 3.9 illustrates an example of image binarization using the proposed adaptive thresholding method comparing with using Ostu's method for a red channel image of an extracted salient region. Note that our proposed adaptive thresholding method is capable of adapting

to local histogram and intensity features in the extracted salient regions instead of the entire image. The original Otsu's method fails to find a good threshold because of a large number of pixels in other regions also having high intensity values in the red channel of the entire image.

As we mentioned before, morphological operations are used to extract the whole area of the hazmat sign from the binarized channel images. First, we use a flood-fill operation to fill holes [123] in the binarized channel images of an extracted salient region A_k^{SR}. A hole is a set of background pixels surrounded by foreground pixels. We use this operation to fill up missing pixels of UN identifier numbers and symbols that are removed due to different colors. Next, we use morphological dilation with a SE_D-pixel diamond shaped structuring element to enlarge the boundaries of foreground areas [123, 124], where SE_D is the size of the diamond shaped structuring element (pixel distance from the origin to the vertex). The shape of the structuring element we used is same diamond as hazmat sign. We use this dilation to merge three separate parts of a whole hazmat sign that divided by the placard holders with two horizontal strips.

$$SE_D = min\left(7, F_{SE} \cdot \mathcal{N}(A_k^{SR})\right), \tag{3.7}$$

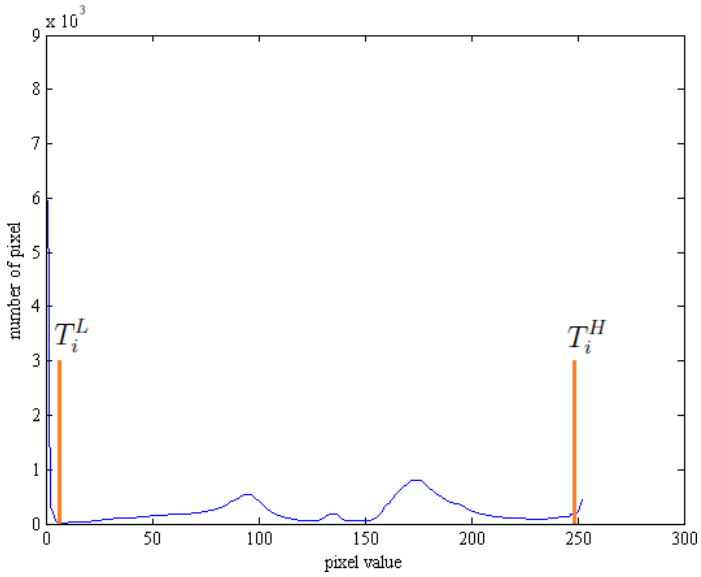
where $\mathcal{N}(A_k^{SR})$ is the total number of pixels in the salient region A_k^{SR} and $F_{SE} = 0.0025\%$ is a factor to determine the size of the diamond shaped structuring element SE_D with respect to the percentage of the total number of pixels in A_k^{SR}, which is empirically determined by searching good values in our experiments.

We use connected component analysis to determine the boundary of the entire hazmat sign in the binarized channel images. We remove small connected components containing less than $T_{CC} = 200$ pixels, which is less than the minimum number of pixels on a hazmat sign in our image datasets. Finally, we obtain closed contours by tracing the exterior boundaries of the connected components [124, 125] in each binarized channel image separately. Table 3.1 lists all the thresholds and parameters we used including empirically obtained ones.

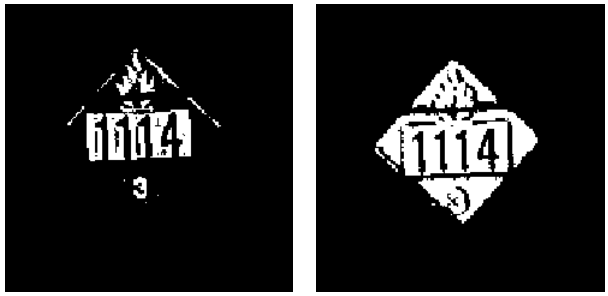
Table 3.1
Thresholds and parameters used in our proposed method. Automatically determined ones are denoted by *.

Symbol	Description	Value
N_i^{MED}	Median of the pixel counts of all bins	*
T_i^L	Low index threshold to clip the histogram	*
T_i^H	High index threshold to clip the histogram	*
T_i^{BW}	Adaptive threshold to binarize channel images	*
SE_D	Size of the diamond shaped structuring element	*
F_B	Factor to determine index thresholds T_i^L and T_i^H	0.05
F_{SE}	Factor to determine the size of the structuring element SE_D	0.0025%
T_{CC}	Threshold to remove small connected components	200
T_e	Threshold for correlation-based matching cost e	1.751

(a) Original image (b) Extracted saliency regions



(c) Histogram of the saliency region in red channel



(d) Otsu's method (e) Proposed method

Fig. 3.9. Example of image binarization using the proposed adaptive thresholding method comparing with using Ostu's method.

3.5.4 Fourier Descriptor Generation

The Fourier Descriptor (FD) describes the shape of an object using a set of the Fourier transform coefficients of the object's contour [104, 105]. Given the extracted contour $c(k)$ has N pixels, numbered from 0 to $N-1$, a set of pixel coordinates describing the contour $c(k)$ can be defined as follows.

$$c(k) = (x(k), y(k)) = x(k) + iy(k), \tag{3.8}$$

where $k = 0, 1, 2, \ldots, N-1$. The Fourier transform of the contour points $c(k)$ generates a set of complex numbers $C(v)$ which are the Fourier descriptors of the contour.

$$C(v) = \mathcal{F}(c(k)) = \frac{1}{N}\sum_{k=0}^{N-1} c(k) exp\left(-\frac{i2\pi vk}{N}\right), \tag{3.9}$$

where $v = 0, 1, 2, \ldots, N-1$. In order to describe the shape of a closed contour generally, the Fourier descriptor have to be modified to make it invariant to translation and scaling [74,109–111]. To achieve translation invariance, the DC Fourier coefficient $C(0)$ is set to zero $C(0) = 0$. All points on the contour are then shifted from its original coordinate to $(0, 0)$. The closed contour represented by the remaining Alternating Current (AC) Fourier coefficients is invariant against translation, but it's still affected by scaling due to the magnitude of each AC coefficient. To achieve scaling invariance, the remaining AC Fourier coefficients $C(v)$ are normalized by $\sqrt{\sum_{v=1}^{N-1} |C(v)|^2}$. The modified Fourier descriptor $C'(v)$ of the extracted contour $c(k)$ are obtained as follows.

$$C'(v) = \begin{cases} 0, & \text{if } v = 0, \\ \frac{C(v)}{\sqrt{\sum_{v=1}^{N-1} |C(v)|^2}}, & \text{if } v \neq 0, \end{cases} \tag{3.10}$$

where $C(v)$ is the original Fourier coefficients. The low frequency components of Fourier descriptors $C'(v)$ contain information about the general shape of the contour while the high frequency components contain finer details. Therefore, the first P

modified AC Fourier descriptors can be used to create an approximate reconstruction $\widehat{b}(k)$ of the original contour points $c(k)$ for contour matching.

$$\widehat{c}(k) = \frac{1}{P}\sum_{v=0}^{P} C'(v) exp\left(\frac{i2\pi vk}{N}\right), \tag{3.11}$$

where $k = 0, 1, 2, \ldots, N-1$.

3.5.5 Correlation-Based Contour Matching[5]

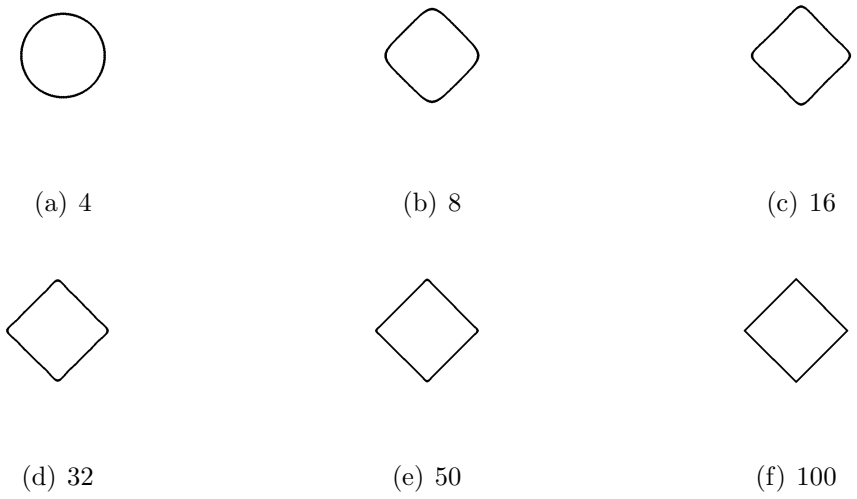
We use the correlation-based contour matching method [74] to locate the border of hazmat signs based on a diamond shaped template. To achieve the rotation and starting point invariance, the correlation-based contour matching using both magnitude and phase information is required for hazmat sign location detection. The modified Fourier descriptors of extracted contours and the template contour can been obtained in previous steps and their magnitude and phase information is used to compute cross-correlation by employing complex conjugate multiplication $\overline{X}Y$. This correlation-based contour matching method is able to achieve translation, scaling, rotation and starting point variances. The cross-correlation $r_{TE}(l)$ between an extracted contour c_E and the template contour c_T is defined as follows.

$$\begin{aligned} r_{TE}(l) &= \int_0^K \overline{c_T(k)} c_E(l+k) \mathrm{d}k \\ &= \sum_{v=0}^{N-1} \overline{C'_T(v)} C'_E(v) exp\left(-\frac{i2\pi vl}{K}\right) \qquad (3.12) \\ &= \mathcal{F}^{-1}\{\overline{C'_T} C'_E\}(v). \qquad (3.13) \end{aligned}$$

By using the first P modified Alternating Current (AC) Fourier descriptors with both magnitude and phase information, this simplified contour matching method is

[5]The work in this section was developed by the author jointly with my colleague Kharittha Thongkor.

able to approximately achieve translation, scaling, rotation and starting point variances. We say "approximately" because we are only using the first few modified Fourier descriptors to describe the shape of the closed contour. In order to determine the appropriate number P of modified AC Fourier descriptors needed for contour matching, we examined the shape variations from a group of reconstructed contours of the template diamond-shaped contour by varying the number of modified AC Fourier descriptors we used. Figure 3.10 illustrates the shape variations of using the first 4, 8, 16, 32, 50 and 100 modified AC Fourier descriptors to reconstruct the template contour of diamond shape. It is shown that using the first 8 modified AC Fourier descriptors is a good approximation of the contour of the diamond shaped template. Using more Fourier descriptors than necessary leads to increasing computational cost with limited additional benefit [106]. Because using more modified AC Fourier descriptors does not significantly improve the matching performance, we only use the first 8 AC Fourier descriptors in our experiments.



(a) 4 (b) 8 (c) 16

(d) 32 (e) 50 (f) 100

Fig. 3.10. The the shape variations of using the first 4, 8, 16, 32, 50 and 100 AC Fourier coefficients.

The modified Fourier descriptors of all the contours are used to match against the ones of the template contour of hazmat signs in Figure 3.7. To decide if an extracted

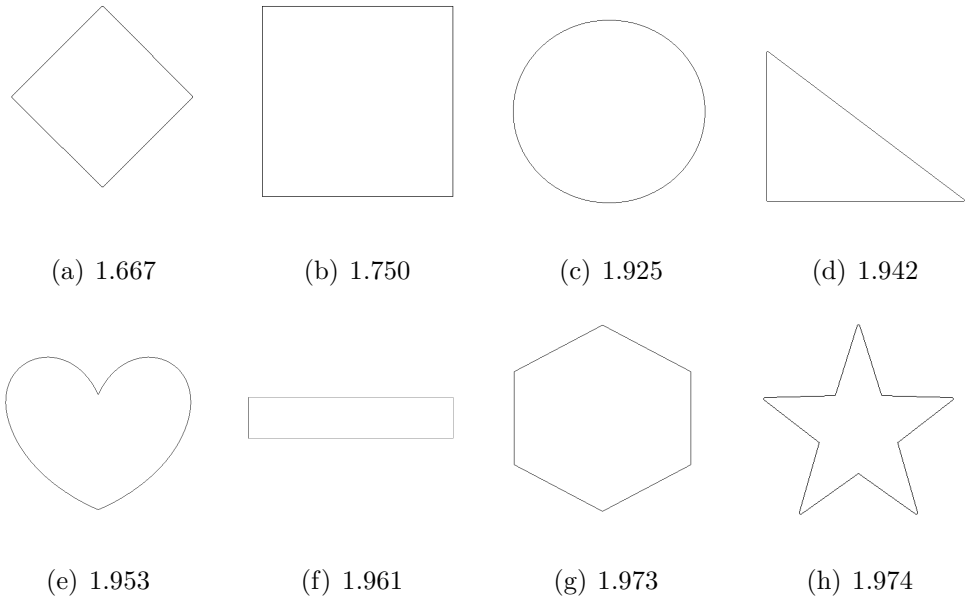
contour c_E is a good match of a hazmat sign, we check the results of a correlation-based matching cost function. The matching cost e is based on the cross-correlation $r_{TE}(l)$ of two modified Fourier descriptors between an extracted contour c_E and the template contour c_T.

$$e = 2 - 2 \max_l |r_{TE}(l)|, \tag{3.14}$$

where $r_{TE}(l)$ is the cross-correlation between an extracted contour c_E and the template contour c_T. If the matching cost e is lower than an empirically obtained threshold T_e, we accept the extracted contour c_E as the border of a hazmat sign that represents the location of that sign in the input image. If the matching cost e is higher than the threshold T_e, we reject the extracted contour c_E and do nothing in the following. In order to determine the threshold T_e, we calculate the correlation-based matching cost e between the contours of some shapes shown in Figure 3.11 and the contour of the diamond shaped template in Figure 3.7. Because the cost of matching a general diamond shape (including the rotation as a square shape) against the diamond shaped template is not greater than 1.750, we then set $T_e = 1.751$. Note that the contours of other shapes in Figure 3.11 are only used to determine the threshold T_e. We keep updating a list of borders representing the sign locations till all the extracted contours in all saliency regions are matched against the template contour. We then obtain the cropped hazmat sign images using the accepted contours in the border list to crop the pixels of hazmat signs from the original image.

3.5.6 Duplicate Sign Removal

To remove duplicate sign candidates from different channel images, we first check all candidates passed the contour matching and estimate their minimal bounding boxes. Any disqualified candidate with the aspect ratio of its bounding box greater than 1.25 will be discarded. We then remove the duplicate sign candidates that correspond to the same sign. This can be done by first dividing all candidates that



(a) 1.667 (b) 1.750 (c) 1.925 (d) 1.942

(e) 1.953 (f) 1.961 (g) 1.973 (h) 1.974

Fig. 3.11. Comparison of the contours of some shapes and their matching costs e.

are overlapped more than 75% into multiple groups and then finding the optimal diamond-shaped box for each group, whose four nodes are closest to the centroid of its group. Each optimal diamond-shaped box is considered to be the location of a detected hazmat sign.

3.5.7 Color Recognition

The HSV color space (Hue, Saturation, Value) is often used for recognizing colors in the Hue (H) channel similar to a color wheel. As Hue (H) varies from 0 to 1, the corresponding colors vary from red through yellow, green, cyan, blue, magenta, and back to red (there are actually red colors both at 0 and 1). As Saturation (S) varies from 0 to 1, the corresponding colors (hues) vary from unsaturated (shades of gray) to fully saturated (no white component). Saturation can be considered as the purity of a color. As Value (V), roughly equivalent to brightness, varies from 0 to 1, the corresponding colors become increasingly brighter. The brightest areas of the value channel correspond to the brightest colors in the original image. Figure 3.12 illustrates the Hue, Saturation, Value of the HSV color space.

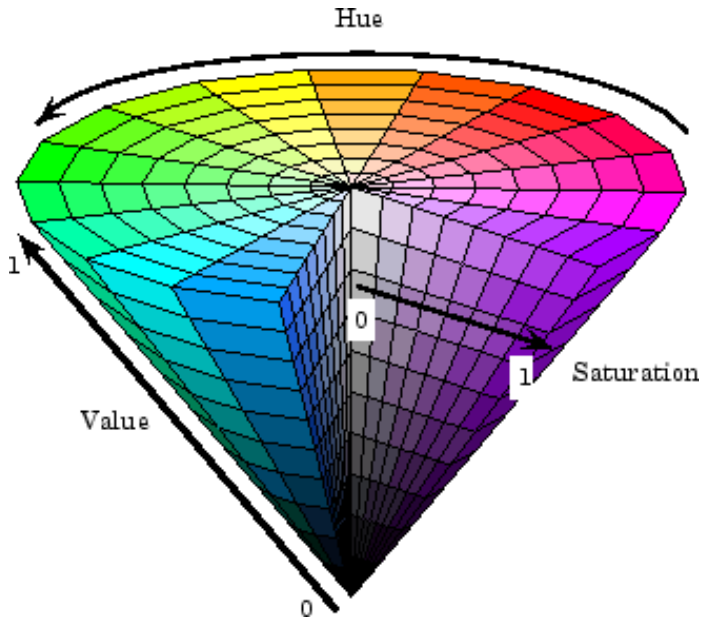


Fig. 3.12. Hazmat sign detection and recognition system.

The color of a hazmat sign can be recognized in HSV color space. We convert a cropped hazmat sign image from RGB to HSV color space and extract the three channel images separately. The white hazmat sign can be first determined from the Saturation (S) and Value (V) channel images of the cropped image. Then other

Table 3.2
The color look-up table based on the 32 uniform distributed hue segments.

Recognized Colors	Red(1)	Orange	Yellow	Green	Blue	Red(2)
Hue Segment Range	0∼0.03125	0.03125∼0.09375	0.09375∼0.25	0.25∼0.5	0.5∼0.75	0.75∼1
Hue Segment Indexes	1	2,3	4∼8	9∼16	17∼24	25∼32

colorful hazmat sign can be recognized from the masked regions of the Hue (H) channel image of the cropped image. This can be done by an image masking method using image thresholding on saturation and value channel images because white's saturation is close to 0 and its value is close to 1. We compute the histograms of the saturation and value channel images of the cropped image and then employ the Otsu's thresholding method [121] to binarize the channel images and obtain two masks of resultant regions whose saturation and value are greater than their thresholds. A combined mask is obtained by AND operations of saturation and value masks and it denotes the mask of color regions in the cropped image since the Saturation (S) and Value (V) channels are both orthogonal to the Hue (H) channel. The masked color regions is used to check if the hazmat sign is white or other color. A hazmat sign is considered as white if the size of the masked color regions is less than 0.4% of the number of pixels in the cropped image, otherwise it is considered as other color in the following.

To determine the color (except white) of the masked color regions, we first define a set of K uniform distributed hue segments by equally dividing the whole range of the hue channel (from 0 to 1). A histogram of K bins of the hue segments is used to characterize the hue distribution of the cropped hazmat sign image. We find the index B_k of the maximum number of pixel counts in the histogram and use it to determine the color (except white) of the hazmat sign by searching B_k in an empirically obtained color look-up table in Table 3.2 based on the K bins of the hue segments. The size of look-up table is determined by the number of hue segments K and we use $K = 32$ in our color recognition method. Some examples of the proposed color recognition

method for white and colorful hazmat signs at 50 feet are illustrated from Figure 3.13 to Figure 3.16 respectively (The hazmat sign images with 4-digit UNID were captured by 5 MP camera on an HTC Wildfire mobile telephone (2592×1952) and the ones with warning text were captured by a 5 MP camera on a Samsung Galaxy Nexus mobile telephone (2592×1944)). The recognized color is used for queuing the mobile database for sign category identification and providing the general guide information based on the 2012 ERG guidebook.

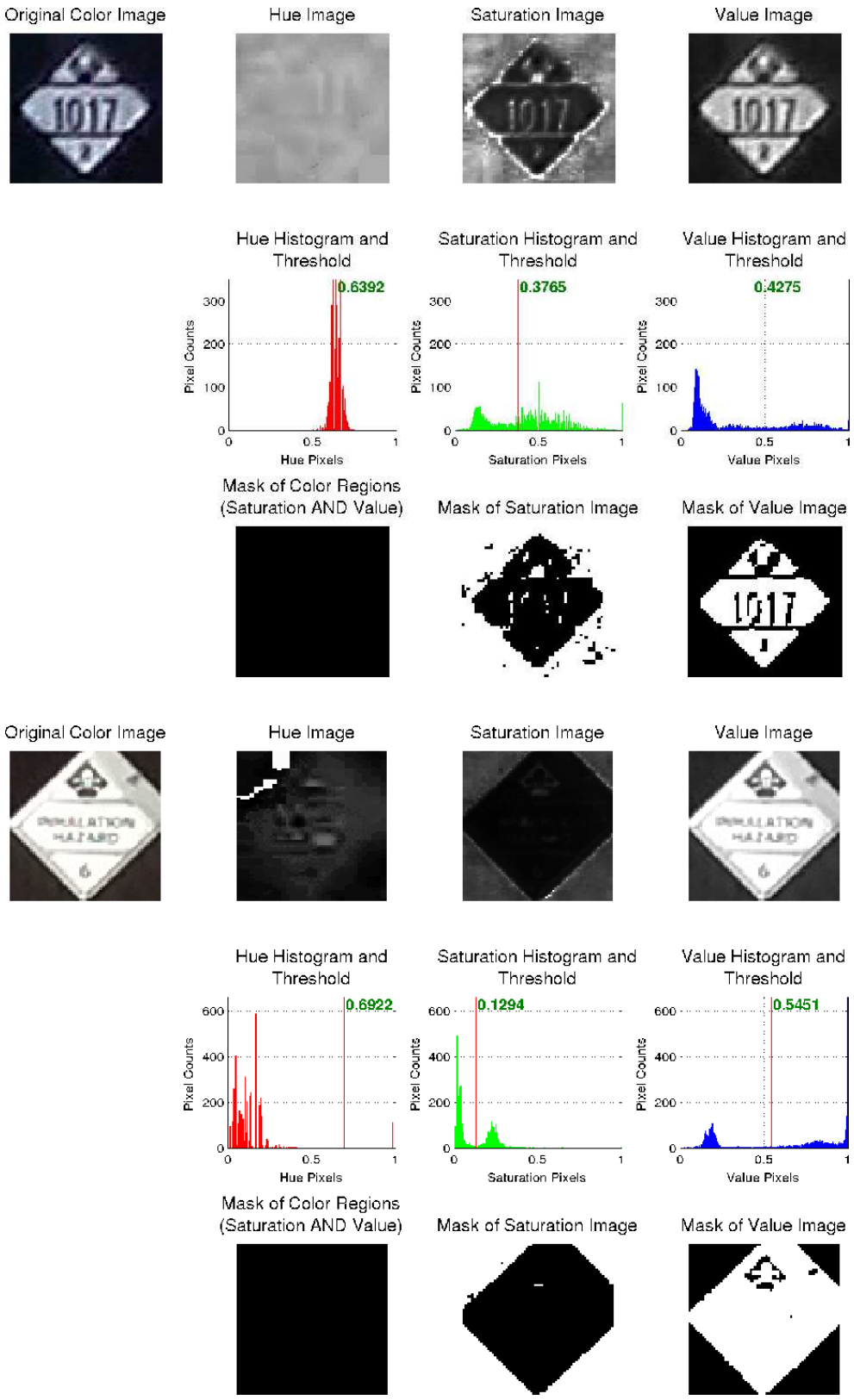


Fig. 3.13. Examples of the proposed color recognition method for two white hazmat signs at 50 feet.

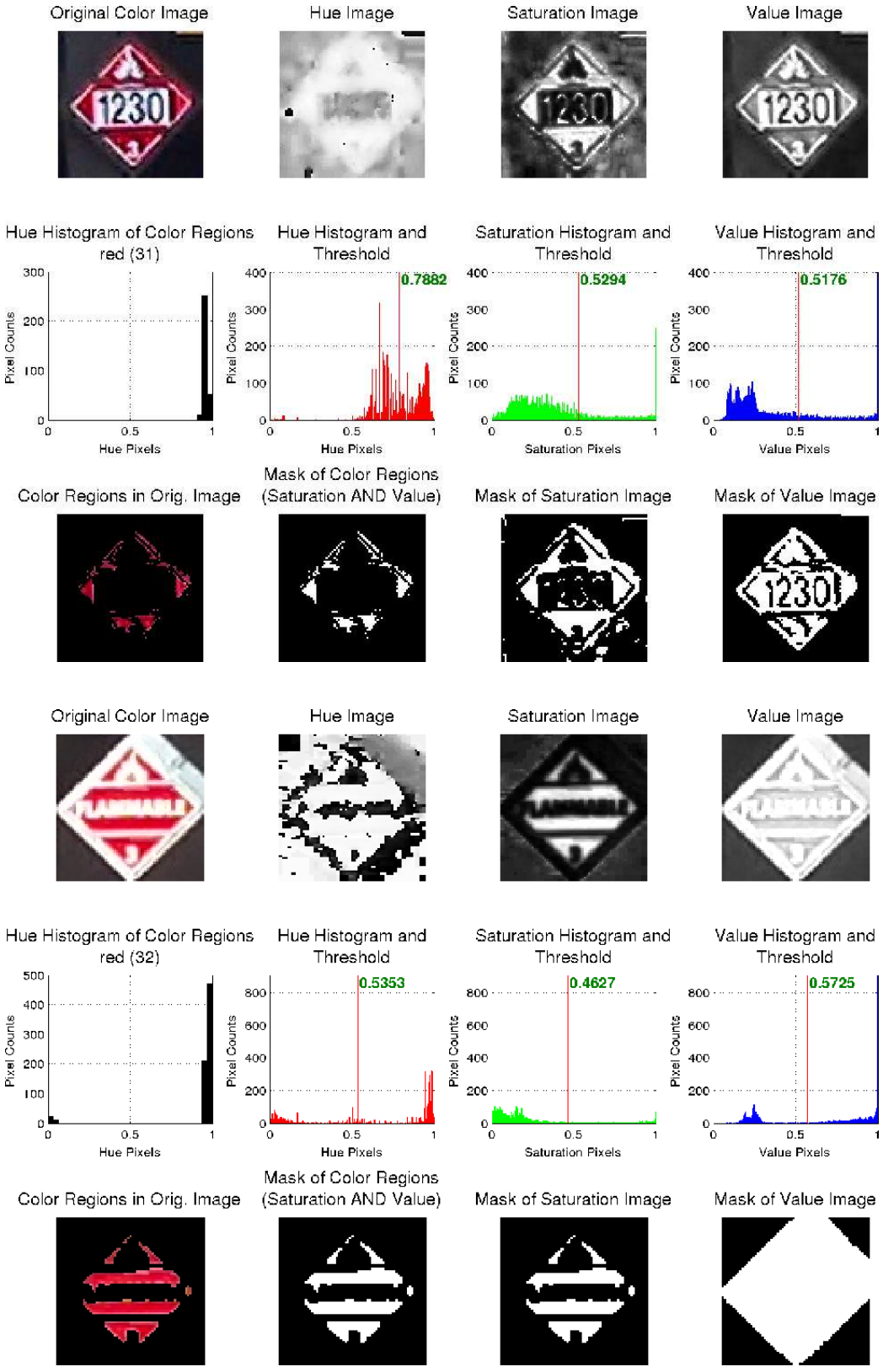


Fig. 3.14. Examples of the proposed color recognition method for two red hazmat signs at 50 feet.

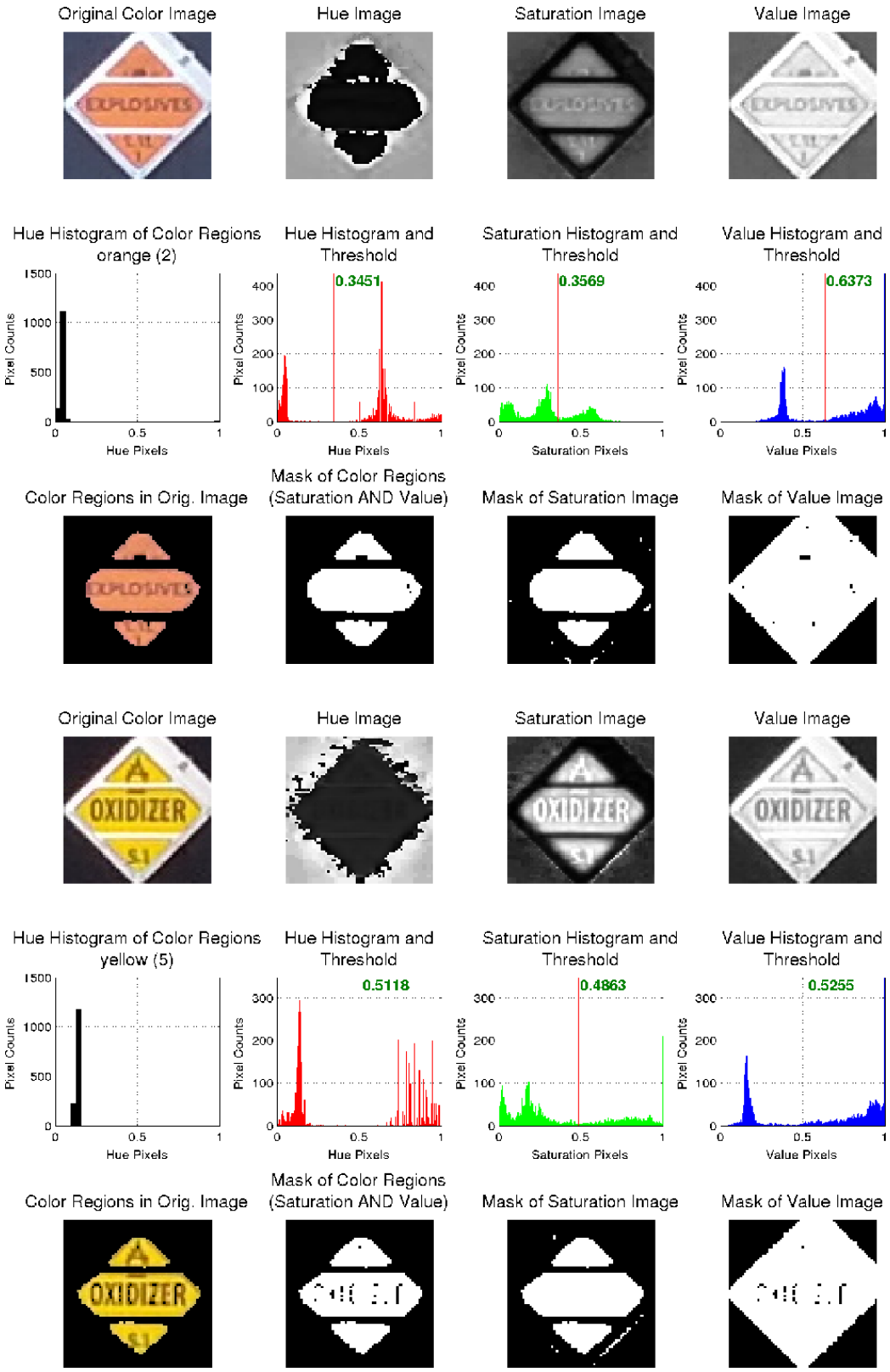


Fig. 3.15. Examples of the proposed color recognition method for orange and yellow hazmat signs at 50 feet.

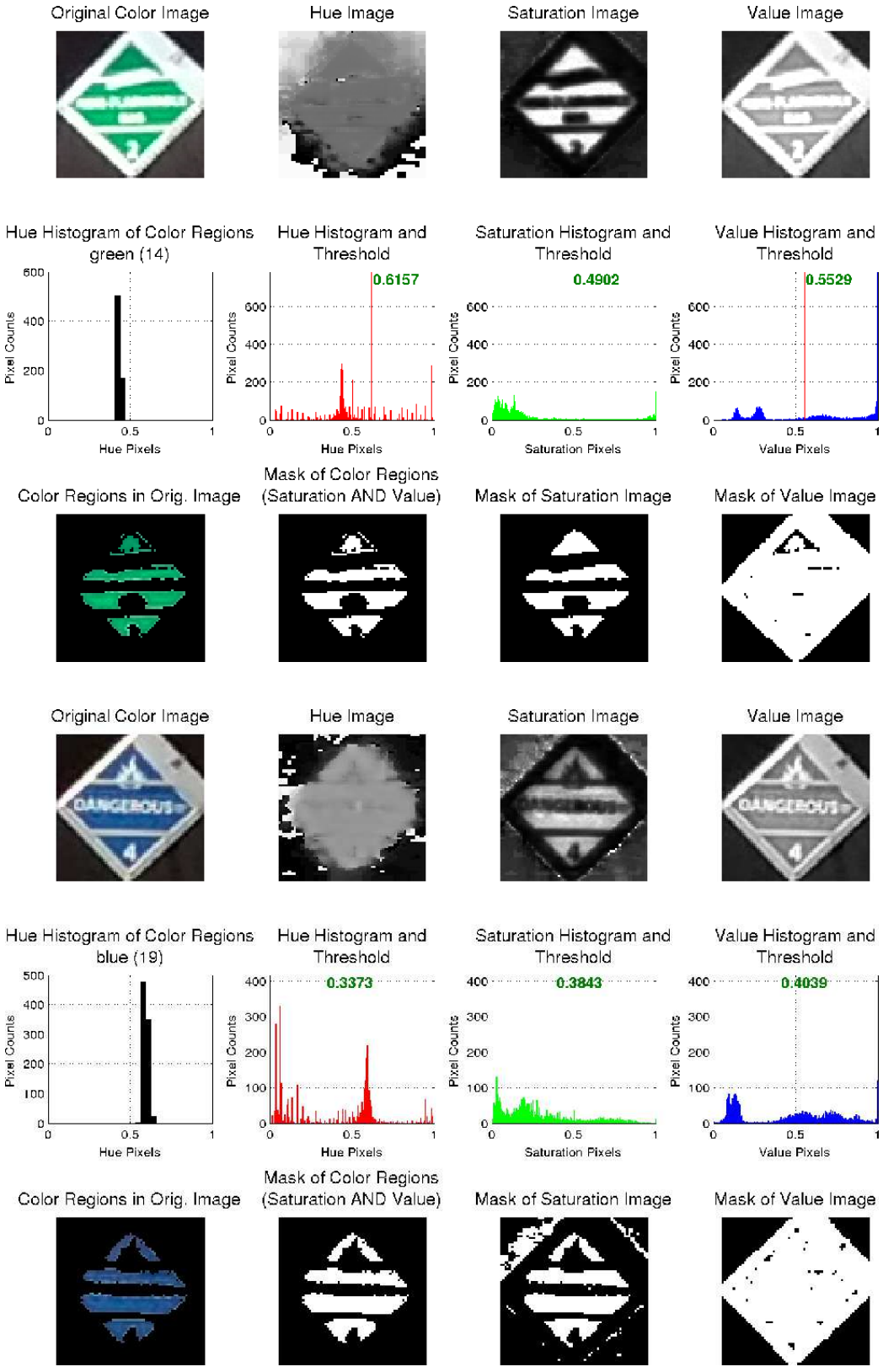


Fig. 3.16. Examples of the proposed color recognition method for green and blue hazmat signs at 50 feet.

3.6 Experimental Results

We did two experiments to investigate the performance and accuracy of our proposed hazmat sign detection and recognition method. The tests were executed on a Galaxy Nexus mobile telephone with a dual-core 1.2GHz CPU and 1GB RAM and a backend server with a quad-core 2.4GHz CPU and 4GB RAM. The first experiment consisted of generating saliency maps using different visual saliency models and evaluating their performance of locating hazmat signs based on ground-truth information. The second experiment consisted of hazmat sign detection and recognition on our image datasets and comparing the results with ground-truth information. The ground-truth information include the image resolution, the number of pixels on the sign, the distance from the camera to the sign, sign color, and sign location in the image.

3.6.1 Image Datasets

Our first image dataset (Dataset-1) consisted of 50 images, each containing one or more hazmat signs in a complex scene (62 hazmat signs in total). The hazmat sign images were captured by a third party under various lighting conditions, distances and perspectives using three different cameras: a 5 MP camera on an HTC Wildfire mobile telephone (2592×1952), an 8.2 MP Kodak Easyshare C813 digital camera (3296×2472), and a 16 MP Nikon Coolpix S800c digital camera (1600×1200) (MP stands for Mega Pixel). Among the 50 images, 23 were reported at 10-50 feet, 23 at 50-100 feet, and 4 at 100-150 feet. Among the 62 hazmat signs, 2 had low resolution, 11 had projective distortion, 8 were blurred, and 6 were shaded. This image dataset contains images of red, yellow, and white hazmat signs. Figure 3.17 illustrates some examples of the first image dataset (Dataset-1) in different conditions.

Our second image dataset (Dataset-2) consisted of 100 images, each containing one or more hazmat signs in a complex scene (111 hazmat signs in total). The hazmat sign images were captured by a third party under various lighting conditions, distances

Fig. 3.17. Examples of the first image dataset (Dataset-1) in different conditions (left to right then top to bottom): low resolution, perspective distortion; blurred sign, shaded sign.

and perspectives using the 16 MP Nikon Coolpix S800c digital camera, including 36 low resolution 2 MP images (1600×1200) and 64 full resolution 16 MP images (4608×3456) from the same camera. Among the 100 images, 22 were reported at 10-50 feet, 35 at 50-100 feet, and 43 at 100-150 feet. Among the 111 hazmat signs, 46 had low resolution, 25 had projective distortion, 12 were blurred, and 17 were shaded. This image dataset contains images of red and white hazmat signs. Figure 3.18 illustrates some examples of the second image dataset (Dataset-2) in different conditions.

Our third image dataset (Dataset-3) consisted of 252 images, each containing only one hazmat sign in a complex scene (252 hazmat signs in total). We use 6 available hazmat signs in different colors for this image dataset, including red, green, blue, orange, yellow, and white. All of them have a warning text in the middle of the signs

Fig. 3.18. Examples of the second image dataset (Dataset-2) in different conditions (left to right then top to bottom): low resolution, perspective distortion; blurred sign, shaded sign.

often used by truck trailer. The images were acquired by us in the outdoor field under various lighting conditions and distances. We took the images at various distances with ground-truth measurement, *i.e.* 10, 25, 50, 75, 100, 125, and 150 feet. The hazmat sign images were captured by us using 3 different cameras: a 5 MP camera on an HTC Wildfire mobile telephone (2592×1952), a 5 MP camera on a Samsung Galaxy Nexus mobile telephone (2592×1944), and a 10 MP Canon PowerShot S95 digital camera (3648×2736) (MP stands for Mega Pixel). At each distance, 36 images were taken by the 3 cameras in both portrait mode and landscape mode (12 images of the 6 hazmat signs in each scene). Among the 252 images, 36 were measured and captured in a straight view at 10 feet, 36 at 25 feet, 36 at 50 feet, 36 at 75 feet, 36 at 100 feet, 36 at 125 feet, and 36 at 150 feet. The 252 hazmat signs have clear appearances without any shape distortion in the images. This image dataset contains images of red, green, blue, orange, yellow, and white hazmat signs. Figure 3.19 and Figure 3.20 illustrate some examples of the 6 signs of the third image dataset (Dataset-

3) at 10 feet in portrait an landscape mode respectively (Images were captured by the 5 MP camera on a Samsung Galaxy Nexus mobile telephone).

The distance information of the first and second image datasets (Dataset-1 and Dataset-2) is visually estimated and thus not reliable. The images were also acquired by a third party in the working field, under various lighting and weather conditions, distances, and perspectives. The distance information of the third image dataset (Dataset-3) is reliable and obtained with ground-truth measurement. The images were acquired by us in the outdoor field under various lighting conditions and distances.

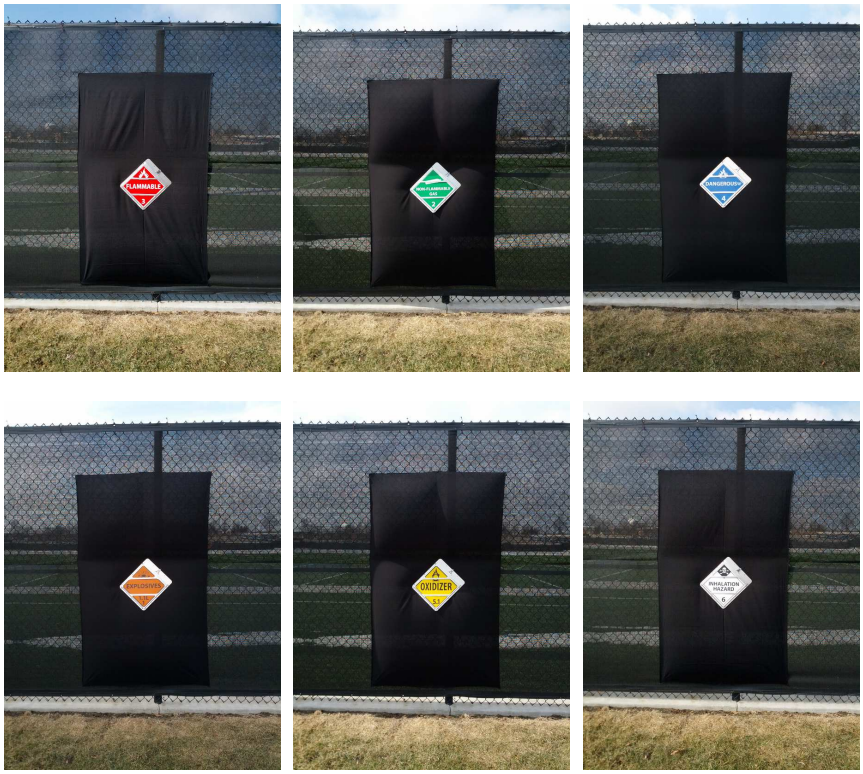


Fig. 3.19. Examples of the 6 signs of Dataset-3 at 10 feet in portrait mode (left to right then top to bottom): red sign, green sign, blue sign; orange sign, yellow sign, white sign.

Fig. 3.20. Examples of the 6 signs of Dataset-3 at 10 feet in landscape mode (left to right then top to bottom): red sign, green sign, blue sign; orange sign, yellow sign, white sign.

Figure 3.21 illustrate some bounding box images for a typical STOP sign and a hazmat sign at the same distance 25, 50, 100, and 150 feet. Table 3.3 shows the relation among the image resolution of a certain camera, the distance from a camera to a hazmat sign, and the number of pixels on a hazmat sign. It also reports the pixel ratio of STOP and hazmat sign by comparing it to a typical STOP sign at each distance. On average, a typical STOP sign contains 5.60 times pixels than a hazmat sign at the same distance.

Fig. 3.21. Examples of bounding box images for a typical STOP sign and a hazmat sign at the same distance 25, 50, 100, and 150 feet.

Table 3.3
The relation among the image resolution, the distance from a camera to a hazmat sign, and the number of pixels on a hazmat sign in the third image dataset (Dataset-3) with comparison to a typical STOP sign.

Camera (Image Resolution)	10 feet	25 feet	50 feet	75 feet	100 feet	125 feet	150 feet
Canon PowerShot S95 (3648×2736)	90312	14450	3444	1458	840	512	364
Hazmat Sign Bounding Box (Pixel Ratio=0.5)	425×425	170×170	83×83	54×54	41×41	32×32	27×27
HTC Wildfire (2592×1952)	57800	9522	2312	1012	578	364	242
Hazmat Sign Bounding Box (Pixel Ratio=0.5)	340×340	138×138	68×68	45×45	34×34	27×27	22×22
Samsung Galaxy Nexus (2592×1944)	56112	9248	2244	1012	578	364	242
Hazmat Sign Bounding Box (Pixel Ratio=0.5)	335×335	136×136	67×67	45×45	34×34	27×27	22×22
Samsung Galaxy Nexus (2592×1944)	298235	50952	12738	5845	3288	1989	1393
STOP Sign Bounding Box (Pixel Ratio=0.82843)	600×600	248×248	124×124	84×84	63×63	49×49	41×41
Pixel Ratio of STOP and Hazmat Sign (Avg=5.60)	5.315	5.510	5.676	5.776	5.689	5.464	5.756

3.6.2 The First Experiment

In the first experiment, we tested our best GCS visual saliency models and 4 state-of-the-art models, including SBVA [25], GBVS [26], IS-DCT-Lab [33], SSA-HFT-IRGBY [35], by hazmat sign image dataset. This experiment consisted of evaluating their performance using a hazmat sign image dataset and scoring the resultant saliency maps in locating hazmat signs based on ground-truth information. We use the first hazmat sign image dataset (Dataset-1) for evaluation and it consists of 50 images and 62 hazmat signs in total.

The saliency models are evaluated in the experiment are: SBVA [25], GBVS [26], IS [33], SSA [35]. We classified the resulting saliency maps into four categories: good, fair, bad, and lost. For each sign, we assigned 3 points for a good saliency map (sign was mostly contained in high salient regions SR_{high}), 2 points for a fair saliency map (sign was mostly contained in middle salient regions SR_{middle}), 1 point for a bad saliency map (sign was mostly contained in low salient regions SR_{low}), and 0 points for a lost saliency map (sign was mostly contained in non-salient regions SR_{non}). The type of salient regions are distinguished by a set of predefined thresholds (The multiples of the average saliency value of a given saliency map based on the Equation 3.2).

$$SR_{high} = \{\bigcup_{(x,y)} S(x,y) | T_{high} \leqslant S(x,y) \leqslant 1\}, \tag{3.15}$$

$$SR_{middle} = \{\bigcup_{(x,y)} S(x,y) | T_{middle} \leqslant S(x,y) < T_{high}\}, \tag{3.16}$$

$$SR_{low} = \{\bigcup_{(x,y)} S(x,y) | T_{low} \leqslant S(x,y) < T_{middle}\}, \tag{3.17}$$

$$SR_{non} = \{\bigcup_{(x,y)} S(x,y) | 0 \leqslant S(x,y) < T_{low}\}, \tag{3.18}$$

$$T_{high} = \frac{4}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} S(x,y), \tag{3.19}$$

$$T_{middle} = \frac{2}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} S(x,y), \tag{3.20}$$

$$T_{low} = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} S(x,y), \tag{3.21}$$

where W and H are the width and height of the saliency map, $S(x,y)$ is the saliency value at position (x,y). Since saliency map is a probability map for predicting the location of eye fixations in a scene, high salient regions are defined as their saliency values are not less than the threshold T_{high}, middle salient regions are between the threshold T_{high} and T_{middle}, low salient regions are between the threshold T_{middle} and T_{low}, and non-salient regions are between the threshold T_{low} and 0. Examples of the four categories of saliency maps (good, fair, bad, lost) with our defined four types of salient regions (high, middle, low, non) are demonstrated from Figure 3.22 to Figure 3.25.

We evaluated above saliency map methods based on average execution times, the distribution of above categories and the calculated score. Table 3.4 shows the results of the visual saliency models in locating hazmat signs. The score of each saliency map method is calculated as the sum of the points assigned for all 62 hazmat signs, which ranges from 0 to 186. Note that the SBVA and the GBVS methods use one color space. Compared with the SBVA and the GBVS methods using one color space, the IS and the SSA methods using one color space have comparable scores, while the IS and

Table 3.4
Average execution time (in seconds), distribution and score of the saliency models (color spaces) in the first image dataset (Dataset-1).

Saliency Map	Time	Good	Fair	Bad	Lost	Score
SBVA(IRGBY)	2.07	28	16	12	6	128
GBVS(IRGBY)	3.36	25	15	15	7	120
IS(Lab)	0.39	27	5	20	10	111
IS(RGB)	0.36	22	7	27	6	107
SSA(Lab)	0.55	33	8	12	9	127
SSA(RGB)	0.53	38	5	8	11	132
IS+SSA(Lab+RGB)	1.83	41	6	8	7	143
GCS(Lab)	0.43	37	10	8	7	139
GCS(RGB)	0.41	28	16	12	6	128
GCS(Lab+RGB)	0.84	52	6	1	3	169

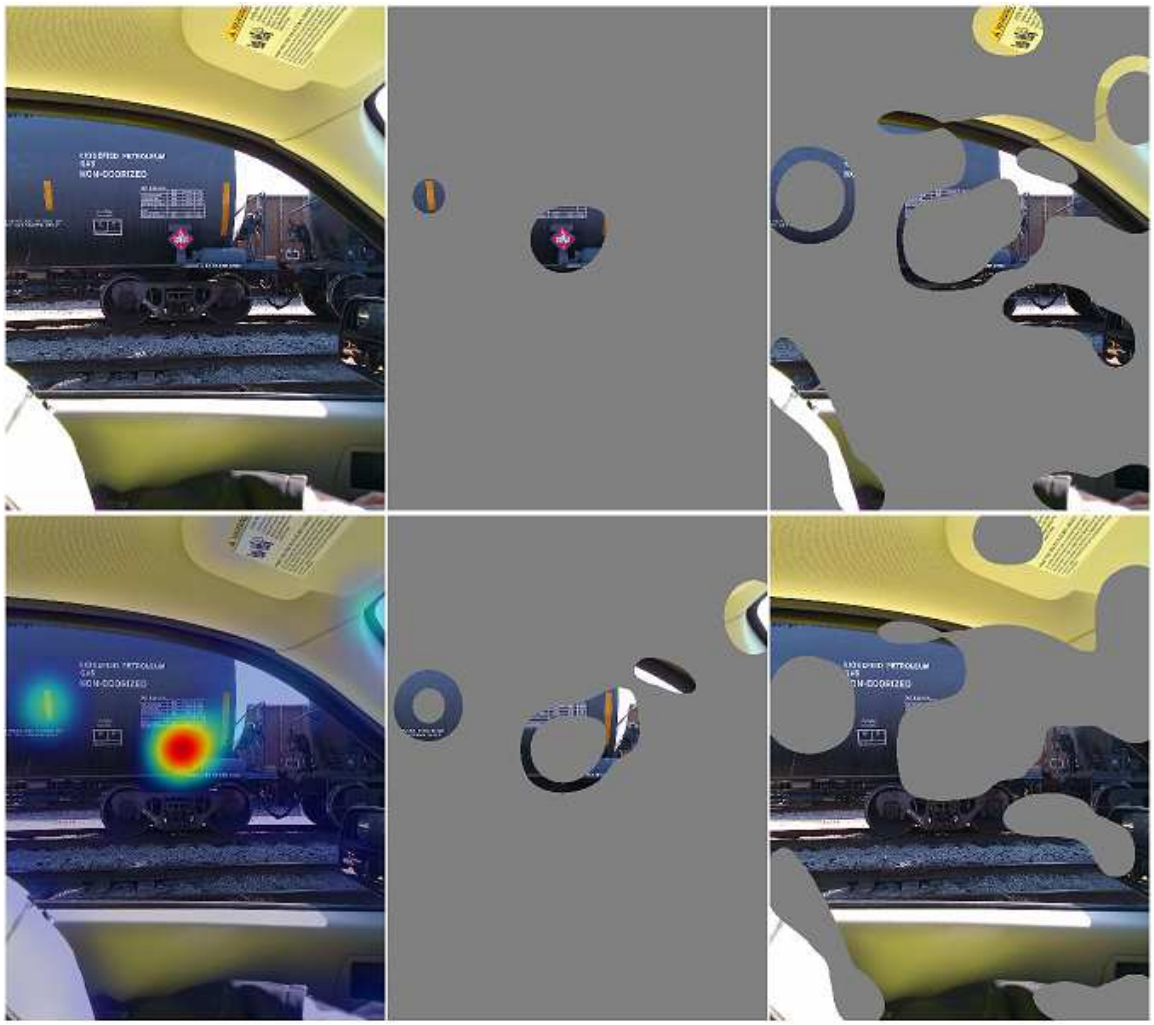


Fig. 3.22. An example of good saliency map with the four types of salient regions (top to bottom then left to right): original image, good saliency map; high salient regions, middle salient regions; low salient regions, non-salient regions.

the SSA methods using two color spaces have higher scores. The GCS(Lab+RGB) and the IS+SSA(Lab+RGB) methods using two color spaces run 2.46 and 1.13 times faster than the SBVA method and 4.0 and 1.84 times faster than the GBVS method respectively. The results verified that the proposed GCS(Lab+RGB) model, combining GCS-FT-Lab and GCS-FT-RGB models, can improve the score of generated

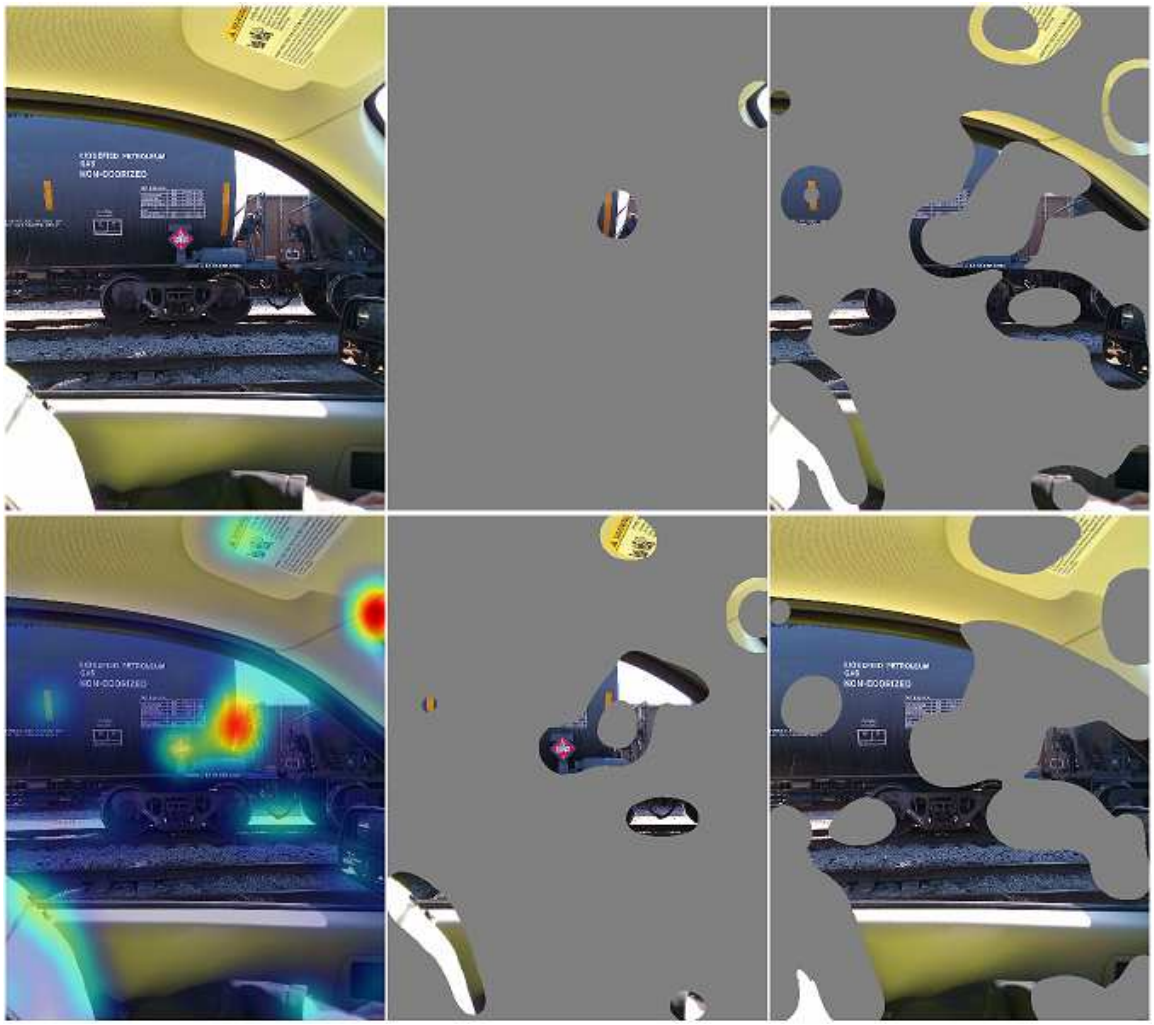


Fig. 3.23. An example of fair saliency map with the four types of salient regions (top to bottom then left to right): original image, fair saliency map; high salient regions, middle salient regions; low salient regions, non-salient regions.

saliency maps, while still running faster than SBVA and GBVS methods. Figure 3.26 and Figure 3.27 illustrate examples of saliency maps from different methods for the same hazmat sign images in portrait mode and landscape mode. Note that the table in the middle indicates the locations of the saliency maps corresponding to which methods.

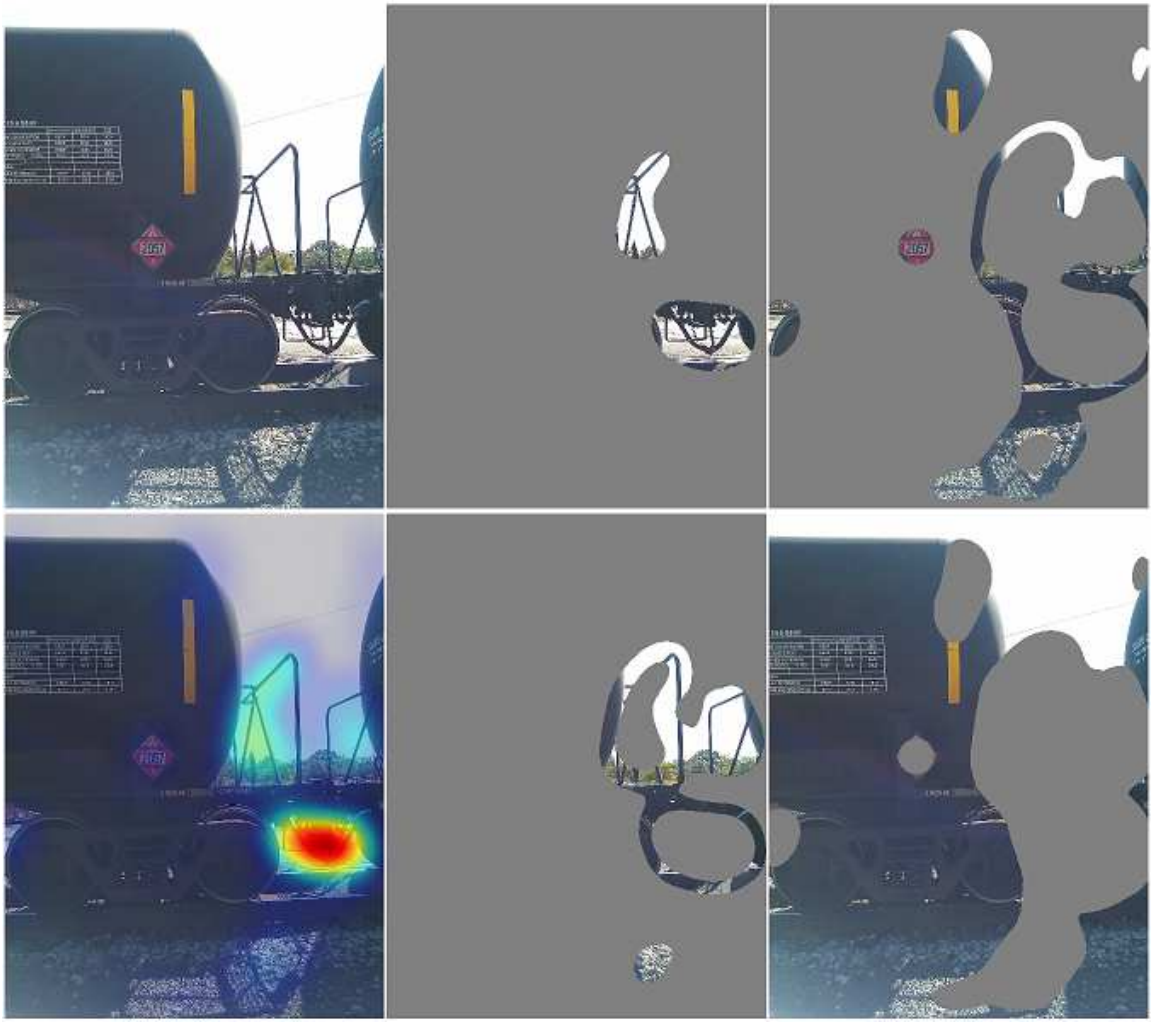


Fig. 3.24. An example of bad saliency map with the four types of salient regions (top to bottom then left to right): original image, bad saliency map; high salient regions, middle salient regions; low salient regions, non-salient regions.

Fig. 3.25. An example of lost saliency map with the four types of salient regions (top to bottom then left to right): original image, lost saliency map; high salient regions, middle salient regions; low salient regions, non-salient regions.

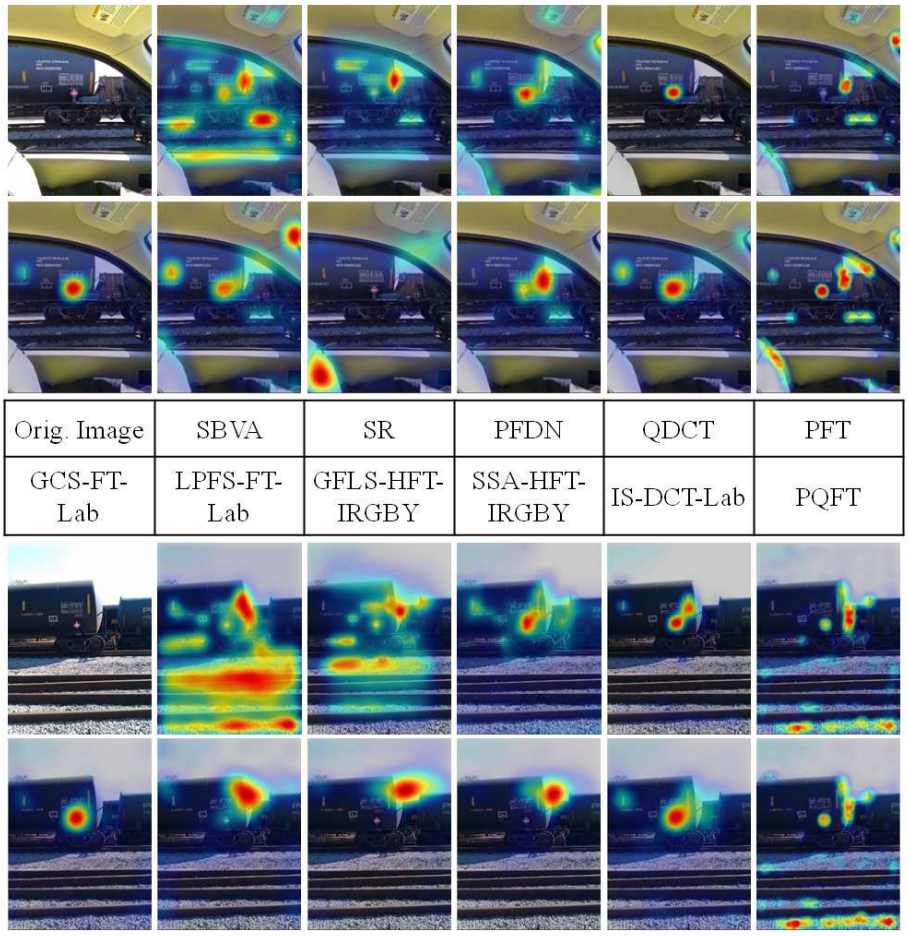


Fig. 3.26. Examples of saliency maps from different methods for two hazmat sign images in portrait mode.

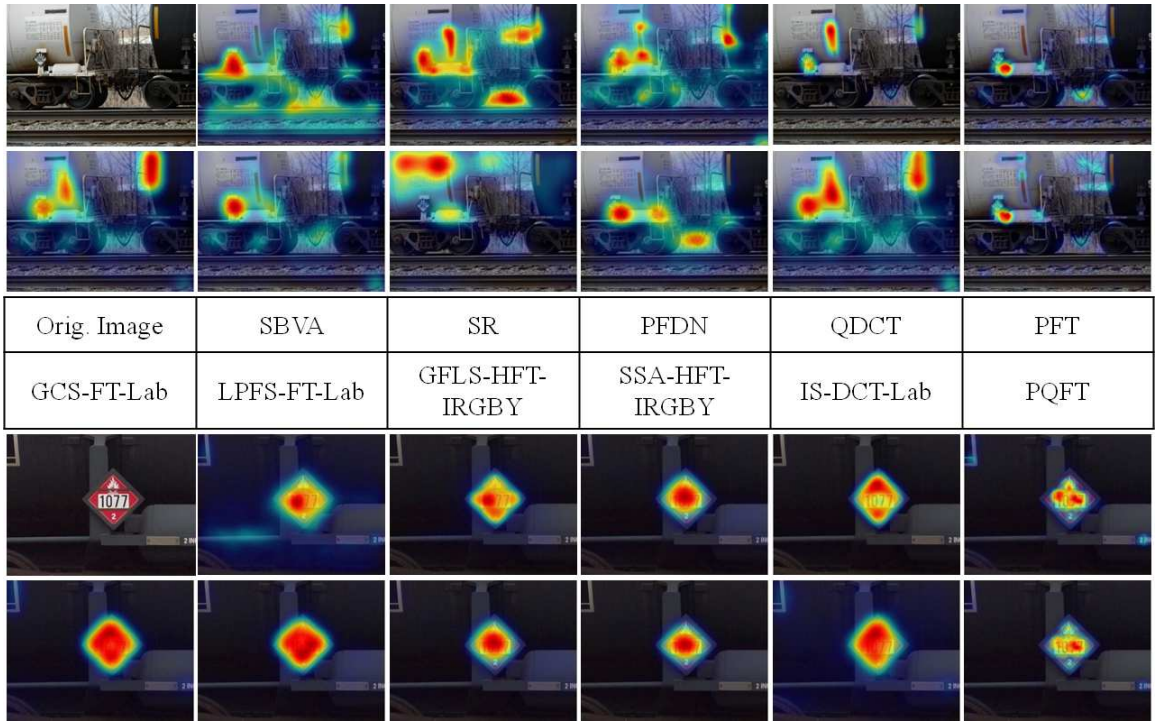


Fig. 3.27. Examples of saliency maps from different methods for two hazmat sign images in landscape mode.

3.6.3 The Second Experiment

In the second experiment, we evaluate the performance of our hazmat sign detection and recognition methods in detecting and recognizing hazmat signs in complex scenes. We employ the following quantitative measurements to evaluate the previous Method 1 and the proposed Method 2.

$$\text{Accuracy} = \frac{\text{The Number of Correct Resultant Signs}}{\text{The Total Number of Signs}}, \tag{3.22}$$

$$\text{Mistakenness} = \frac{\text{The Number of False Positive Objects}}{\text{The Total Number of Signs}}, \tag{3.23}$$

$$\text{Sign-Coverage} = \frac{\text{The Number of Signs Covered in Extracted Salient Regions}}{\text{The Total Number of Signs}}, \tag{3.24}$$

$$\text{Pixel-Usage} = \frac{\text{The Number of Pixels Used in Extracted Salient Regions}}{\text{The Total Number of Pixels in The Image}}. \tag{3.25}$$

Table 3.5 illustrates the performance of the generated saliency maps and salient region extraction methods in terms of the pixel usage and sign coverage in the extracted salient regions of complex scenes. For the three image datasets, the previous Method 1 using four saliency maps obtains the average pixel usage 13.81% and the average sign coverage 96.24%, while the proposed Method 2 using two saliency maps achieves the average pixel usage 10.98% and the average sign coverage 97.41%. Hazmat sign image analysis focusing on the extracted salient regions can achieve good sign coverage and further speed up the overall image analysis process by using only a small portion of pixels, instead of using the entire pixels in an image.

Table 3.5
The pixel usage and sign coverage in the extracted salient regions for the three image datasets.

Proposed Method	Dataset-1 Pixel Usage	Dataset-1 Sign Coverage	Dataset-2 Pixel Usage	Dataset-2 Sign Coverage	Dataset-3 Pixel Usage	Dataset-3 Sign Coverage	Average Pixel Usage	Average Sign Coverage
IS+SSA(Lab+RGB) Sal. Maps	15.29%	98.39%(61/62)	14.63%	95.50%(106/111)	11.52%	96.03%(242/252)	13.81%	96.24%(409/425)
GCS(Lab+RGB) Sal. Maps	12.52%	98.39%(61/62)	11.73%	96.40%(107/111)	8.70%	97.62%(246/252)	10.98%	97.41%(414/425)

The accuracy of sign location detection is significantly related to the image resolution of a certain camera, the distance from a camera to a hazmat sign, and the number of pixels on a hazmat sign. We then determine the color recognition accuracy based on how many signs were correctly color recognized after a successful sign location detection. Note that we used the recognized color inside the sign, not text or UN identifier numbers, queuing the mobile database for sign category identification and providing the general guide information. Therefore the overall accuracy is equivalent to the color recognition accuracy in our experiments.

Table 3.6 illustrates the image analysis results of the proposed methods for the first image dataset (Dataset-1). The previous Method 1 using the IS+SSA(Lab+RGB) model has the location detection accuracy 64.52% and the color recognition accuracy 45.16% for all 62 hazmat signs. The proposed Method 2 using our GCS(Lab+RGB) model obtains the location detection accuracy 67.74% and the color recognition accuracy 61.29%, while the same method without using saliency maps yields 56.45% and 50.00% respectively. Table 3.7 demonstrates the image analysis results of the proposed methods for the second image dataset (Dataset-2). The previous Method 1 using the IS+SSA(Lab+RGB) model has the location detection accuracy 40.54% and the color recognition accuracy 29.73% for all 111 hazmat signs. The proposed Method 2 using our GCS(Lab+RGB) model obtain the location detection accuracy 54.05% and the color recognition accuracy 53.15%, while the same method without using saliency maps yields 43.24% and 42.34% respectively. Compared with our previous Method 1 using the IS+SSA(Lab+RGB) model with four saliency maps, our proposed Method 2 using our GCS(Lab+RGB) model with two saliency maps has higher accuracy of sign location detection in general. Our experimental results confirmed that the proposed visual saliency based image analysis methods can increase the accuracy of sign location detection and reduce the false positive (FP) objects.

For our third image dataset (Dataset-3), Table 3.8, Table 3.9 and Table 3.10 show more image analysis results of the proposed methods at variant distances for the third image dataset (Dataset-3), including the mistakenness of false positive (FP) objects,

Table 3.6
Image analysis results for the first image dataset (Dataset-1).

Proposed Method	Total Signs	FP Object Extracted	FP Object Mistakenness	Location Detected	Location Accuracy	Color Recognized	Color Accuracy	Overall Accuracy
Method 1 IS+SSA(Lab+RGB) Sal. Maps	62	10	16.13%	40	64.52%	28	45.16%	45.16%
Method 2 without Sal. Maps	62	32	51.61%	35	56.45%	31	50.00%	50.00%
Method 2 GCS(Lab+RGB) Sal. Maps	62	21	33.87%	42	67.74%	38	61.29%	61.29%

Table 3.7
Image analysis results for the second image dataset (Dataset-2).

Proposed Method	Total Signs	FP Object Extracted	FP Object Mistakenness	Location Detected	Location Accuracy	Color Recognized	Color Accuracy	Overall Accuracy
Method 1 IS+SSA(Lab+RGB) Sal. Maps	111	24	21.62%	45	40.54%	33	29.73%	29.73%
Method 2 without Sal. Maps	111	81	72.97%	48	43.24%	47	42.34%	42.34%
Method 2 GCS(Lab+RGB) Sal. Maps	111	45	40.54%	60	54.05%	59	53.15%	53.15%

the location detection accuracy, the color recognition accuracy, and the overall process accuracy. The previous Method 1 using the IS+SSA(Lab+RGB) model has the average location detection accuracy 56.75% and the average color recognition accuracy 19.05% for all 252 hazmat signs. The proposed Method 2 using our GCS(Lab+RGB) model achieves the average location detection accuracy 96.83% and the average color recognition accuracy 90.87%, while the same method using saliency maps obtains 87.70% and 82.54% respectively. The previous Method 1 has high location accuracy at short distances but decreases after 50 feet and fails to detect sign location after 125 feet. The proposed Method 2 achieve relatively consistent location accuracy at all distances from 10 feet to 150 feet, because we used the adaptive contour extraction method within more accurate saliency regions and the robust contour matching method based on Fourier descriptors.

For the overall process of our hazmat sign image analysis system, the average execution time of the previous Method 1 and the one of the proposed Method 2 are 5.88 and 5.28 seconds respectively for the first image dataset (Dataset-1), 18.95 and 16.45 seconds respectively for the second image dataset (Dataset-2), while 10.24 and 8.98 seconds respectively for the third image dataset (Dataset-3). The average execution time of the proposed Method 2 without using saliency maps are 8.49, 26.52, and 14.36 seconds for the three image datasets respectively. Our experimental results verified that the proposed visual saliency based methods can speed up the overall image analysis process. With better location detection accuracy and color recognition accuracy, the proposed Method 2 using the GCS(Lab+RGB) model is faster than the previous Method 1 using the IS+SSA(Lab+RGB) model and more suitable for practical applications and uses.

Table 3.8
Image analysis results of Method 1 using four saliency maps for the third image dataset (Dataset-3).

Method 1 IS+SSA(Lab+RGB) Sal. Maps	Total Signs	FP Object Extracted	FP Object Mistakenness	Location Detected	Location Accuracy	Color Recognized	Color Accuracy	Overall Accuracy
10 feet	36	8	22.22%	36	100.00%	10	27.78%	27.78%
25 feet	36	4	11.11%	36	100.00%	16	44.44%	44.44%
50 feet	36	2	5.56%	34	94.44%	10	27.78%	27.78%
75 feet	36	1	2.78%	23	63.89%	7	19.44%	19.44%
100 feet	36	1	2.78%	11	30.56%	3	8.33%	8.33%
125 feet	36	4	11.11%	3	8.33%	2	5.56%	5.56%
150 feet	36	4	11.11%	0	0.00%	0	0.00%	0.00%
Average	252	24	9.52%	143	56.75%	48	19.05%	19.05%

Table 3.9
Image analysis results of Method 2 without using saliency maps for the third image dataset (Dataset-3).

Method 2 without using Sal. Maps	Total Signs	FP Object Extracted	FP Object Mistakenness	Location Detected	Location Accuracy	Color Recognized	Color Accuracy	Overall Accuracy
10 feet	36	4	11.11%	33	91.67%	29	80.56%	80.56%
25 feet	36	50	138.89%	26	72.22%	26	72.22%	72.22%
50 feet	36	3	8.33%	34	94.44%	33	91.67%	91.67%
75 feet	36	5	13.89%	32	88.89%	30	83.33%	83.33%
100 feet	36	3	8.33%	34	94.44%	32	88.89%	88.89%
125 feet	36	9	25.00%	30	83.33%	28	77.78%	77.78%
150 feet	36	3	8.33%	32	88.89%	30	83.33%	83.33%
Average	252	77	30.56%	221	87.70%	208	82.54%	82.54%

Table 3.10
Image analysis results of Method 2 using two saliency maps for the third image dataset (Dataset-3).

Method 2 GCS(Lab+RGB) Sal. Maps	Total Signs	FP Object Extracted	FP Object Mistakenness	Location Detected	Location Accuracy	Color Recognized	Color Accuracy	Overall Accuracy
10 feet	36	0	0.00%	36	100.00%	32	88.89%	88.89%
25 feet	36	1	2.78%	36	100.00%	36	100.00%	100.00%
50 feet	36	0	0.00%	36	100.00%	35	97.22%	97.22%
75 feet	36	1	2.78%	36	100.00%	34	94.44%	94.44%
100 feet	36	7	19.44%	33	91.67%	31	86.11%	86.11%
125 feet	36	20	55.56%	33	91.67%	30	83.33%	83.33%
150 feet	36	19	52.78%	34	94.44%	31	86.11%	86.11%
Average	252	48	19.05%	244	96.83%	229	90.87%	90.87%

Discussion

¿From the experiments our accuracy ranged from approximately 90% for images taken is a controlled test environment to approximately 60% for image acquired in a more "typical" operating scenario. The accuracy we obtained in a more typical operating scenario would not be acceptable for many situations. The overall accuracy of our hazmat sign image analysis system is affected by the location and color recognition accuracy. The accuracy of sign location detection is significantly related to the image resolution of the camera, the distance from the camera to a hazmat sign, and the number of pixels forming the hazmat sign. As mentioned above, we show the relation between these factors in Table 3.3. The location detection accuracy of the first Dataset-1 67.74% and the second Dataset-2 54.05% were lower than the average one of the third Dataset-3 96.83%, because there are a large number of blurred, low resolution, and perspective distorted hazmat signs contained in the first Dataset-1 and the second Dataset-2. Note that hazmat signs in the third Dataset-3 were acquired in a controlled test environment without any shape distortion in the images. The location detection accuracy deteriorates due to the loss of boundary contours for blurred and low resolution hazmat signs and poor correlation in contour matching for perspective distorted hazmat signs.

The color recognition accuracy is based on how many signs were correctly color recognized after a successful sign location detection. The color recognition accuracy of the first Dataset-1 61.29% and the second Dataset-2 53.15% were also lower than the average one of the third Dataset-3 90.87%, because the color recognition accuracy will not exceed the previous location detection accuracy. The color recognition accuracy is also degraded by the absence of color calibration for hazmat sign images, especially for shaded signs, which cause our color recognition method to misidentify the sign color.

The location detection accuracy could be improved by using supper-resolution image reconstruction methods [126, 127] to refine hazmat sign images at the step of

image preprocessing. The color recognition accuracy could be increased by employing proper color calibration methods [51] at the step of image acquisition. Therefore we could further improve the overall accuracy of our hazmat sign image analysis system.

4. ERROR CONCEALMENT FOR SCALABLE VIDEO CODING

A Scalable video coding (SVC) decoder typically requires that the base layer frames be delivered almost error-free and uses them to decode the enhancement layer frames. Due to the nature of dynamic and lossy channels used for video delivery (particularly wireless channels), video bitstreams transmitted over packet networks usually experience isolated and burst packet losses [17]. An accurate distortion model for the effect of different packet loss patterns on the encoded video was proposed in [128]. It confirmed that a burst packet loss produces a larger distortion than an equal number of isolated packet losses. Moreover, once errors occur in video bitstreams, they are prone to propagate from one frame to another due to motion-compensated prediction used in SVC codec. These effects can result in severe visual quality degradation of the decoded frames.

Error concealment (EC) is an effective scheme for error recovery, which imposes small complexity on the decoder and provides a flexible solution to the above problems [129, 130]. By the use of error concealment methods, damaged regions can be reconstructed from the correctly received neighboring regions. Due to the layered structure of SVC, it is advantageous to recover the damaged frames in one layer using the available frames in other layers. It has been shown that one can exploit the spatial and temporal correlations of video frames between different layers to improve the performance of single layer error concealment [18].

Slice structuring [130] is a useful strategy to reduce error propagation from a damaged slice/packet to subsequent slices/packets from burst packet losses. Slice interleaving (SI) [131] and flexible macroblock ordering (FMO) [132] are two common slice structuring schemes. Interleaving approach has been exploited for slice

structuring and packetization. A near-optimal packet interleaving method was proposed in [133] based on an optimization criteria in terms of temporal neighbor packet distance. Another packetization method was introduced in [134] based on optimal packetization masks, which aims to simultaneously maximize the intra-partition distance and distribute neighboring coefficients equally among different packets. The FMO technique has been employed to independently assign each macroblock (MB) of a frame to a certain slice group (SG) by a macroblock allocation map (MBAmap). The H.264/AVC video coding standard specifies seven types of FMO to support error resilience [132]. FMO Type 1 is also known as scattered or dispersed slices. The effect of error propagation between frames has been investigated in [135] and a more suitable MBAmap with a reduced effect of error propagation can be generated based on the evaluation of each macroblock's importance. In [136], an adaptive MBAmap updating scheme is proposed to reduce the computational cost of FMO and a slice matching error concealment method is also introduced. In [137], a new FMO method was proposed by solving an optimization problem of optimal MB labeling for burst packet loss resilience.

In this chapter, a two-layer spatial-temporal SVC system is developed for inter-layer error concealment. The enhancement layer has high spatial resolution at high frame rate (e.g. 30 fps) and the base layer has low resolution at low frame rate (e.g. 15 fps). It is assumed that the packet delivery of the base layer is loss-prone the same way as the enhancement layer. In this scenario, three inter-layer error concealment methods are proposed using two new approaches. (1) Motion vector averaging using adaptively averaging over multiple types of motion vectors in different layers for the recovery of lost motion vectors. (2) Slice interleaving utilizing an optimum ordering technique to make the average distance between two contiguous slices as far as possible. The proposed error concealment method is capable of decoding the SVC bitstreams under burst packet losses and reconstructing the damaged frames with enhanced visual quality. The effect of burst packet losses and error propagation on

video frames in both layers is investigated regarding two existing and three proposed error concealment methods.

4.1 Error Concealment Methods

4.1.1 Conventional Error Concealment Methods

The SVC reference codec, Joint Scalable Video Model (JSVM), introduced four non-normative error concealment (EC) methods [138] to address the problem of error recovery. (1) Picture copy (PC): Each pixel value of the concealed frame is copied from the corresponding pixel of the first frame in the reference frame list 0. (2) Temporal direct motion vector generation (TD): This predicts a missing frame using two reference frame lists and generates the desired missing motion vectors by scaling the motion vectors inferred from its neighboring reference frames. (3) Motion and residual upsampling or base layer skip (BLSkip): This conceals a lost enhancement layer frame from the predicted P- or B-frames. The residuals and motion vectors of the base layer will be up-sampled to higher resolution for the enhancement layer. (4) Reconstruction base layer upsampling (RU): The base layer frame is reconstructed and up-sampled using a 6-tap H.264/AVC filter for the lost enhancement layer frame. In addition, a new intra-layer method was introduced in [18]. (5) Motion copy (MC): The reconstruction of the last key frame is re-used as the reference. Motion vectors are re-generated by copying the motion field of the last key frame. Single-layer EC methods include FC, TD and MC, while inter-layer EC methods include BLSkip and RU. The experimental results in [18] concluded that the BLSkip-based method is a desirable SVC EC tool.

4.1.2 Related Work

Motion vector error concealment has been an active research area for many years. A block-based motion vector extrapolation (MVE) method was proposed in [139].

Some MVs of correlated MBs in the previous frame are first extrapolated to the current damaged frame and then the lost MV of the damaged MB is replaced by the best MV of the motion extrapolated MB with the largest overlapped area. A pixel-based MVE (PMVE) method was introduced [140] by extending the block-based MVE method [139] to the pixel level. A hybrid MVE (HMVE) scheme was proposed in [141] based on the pixel-based and block-based MVE, which is able to discard the wrongly extrapolated MVs in order to obtain more accurate MV. In [142], a block-based motion projection (MP) approach was proposed to reconstruct the lost MV of the damaged block based on its qualified temporal blocks' MVs and spatial neighbors' MVs. In general, block-based MVE and MP methods are similar in terms of that they all used MVs from the projected blocks in previous frame and select the best MV of the block with the largest overlapped area. But MP employs a post-processing stage and median filtering is capable of refining the reconstructed MV field.

The visual quality of the error concealed regions can be further improved with the help of slice interleaving. It is aimed at spreading contiguous slices over different packets against packet losses, so that damaged regions can be surrounded by some correctly received regions. A simple slice interleaving approach was used in [131], where each slice consists of disjoint single lines of macroblocks in a frame. In [128], a packet interleaver was presented to interleave the packets before transmission and cope with burst packet losses, where packets are first loaded into the block interleaver in rows and are transmitted by columns. A distance-based slice interleaving method [143] was proposed to rearrange independently decodable slices of consecutive frames into packets according to an optimal interleaving structure for packetization. Each slice is interleaved by achieving the maximum minimal distance between contiguous slices.

4.1.3 Proposed Error Concealment Methods

In burst packet loss environments three inter-layer error concealment methods are proposed using two new approaches: (1) adaptively averaging over multiple types of motion vectors in different layers and (2) slice interleaving by an optimum ordering technique.

Motion Vector Averaging

We propose a new inter-layer motion vector averaging approach to reconstruct lost motion vectors. It uses a 4x4 block for the base layer and an 8x8 block for the enhancement layer as the basic concealment units. As shown in Figure 4.1, this inter-layer motion vector averaging approach exploits the spatial and temporal correlations of motion vectors between the two layers (co-located motion vectors MV_e^{BL} and $MV_{o/e}^{EL}$, where MV_o and MV_e denote motion vectors for a specific odd and even frame number respectively) and also uses a predictive motion vector MV_{Pred}^{EL} and a median motion vector $MV_{Med}^{EL/BL}$. MV_{Pred}^{EL} is a weighted average of the motion vectors of four projection-overlapped blocks in a reference frame f_r^{EL} and each weight $w(i)$ is the ratio of the size of each overlapped portion to the projection block size. $MV_{Med}^{EL/BL}$ is obtained based on the MVE [139] estimated ($MV_{MVE}^{EL/BL}$) and its neighbors' ($MV_{Nb}^{EL/BL}$) motion vectors in the same EL/BL frame $f_c^{EL/BL}$ respectively. Our method recovers a lost motion vector in one layer by adaptively averaging over multiple types of motion vectors in two layers using a multi-hypothesis parameter $\alpha \in [0, 1]$. Note that $\lfloor * \rceil$ represents the rounding function of $*$ to the nearest integer, s denotes the s-neighborhood adjoining blocks $s \in \{4, 8\}$, and $MV_{Med} = Median\{MV(k)\} = (Median\{MV^x(k)\}, Median\{MV^y(k)\}) = (MV_{Med}^x, MV_{Med}^y)$ for $k \in \{1, 2, \cdots, s\}$.

Base Layer (BL): The lost motion vector MV_e^{BL} in the BL current frame f_c^{BL} can be recovered in two cases. In case 1, if MV_e^{BL} is lost but MV_e^{EL} is correctly received, MV_e^{BL} can be reconstructed by adaptively averaging over two synthetic motion vectors. One is an aggregative motion vector by combining an approximate

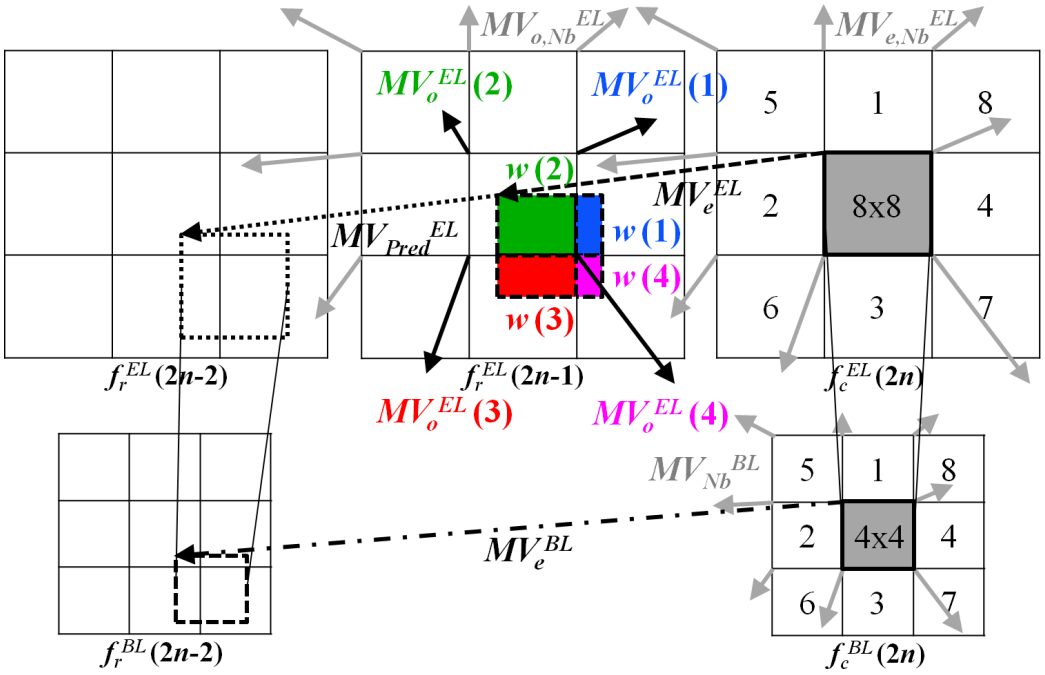


Fig. 4.1. The proposed inter-layer motion vector averaging approach using adaptively averaging over multiple types of motion vectors in two layers.

motion vector $\frac{1}{2}MV_e^{EL}$ in the EL corresponding frame f_c^{EL} and a predictive motion vector $\frac{1}{2}MV_{Pred}^{EL}$ in the EL reference frame f_r^{EL}. The other is a median motion vector MV_{Med}^{BL} based on the MVE estimated and s neighbors' motion vectors in the same BL frame. In case 2, if MV_e^{EL} and MV_e^{BL} are both lost, MV_e^{BL} can be reconstructed using the median median motion vector MV_{Med}^{BL} in the same BL frame.

BL Case 1: If MV_e^{BL} is lost but MV_e^{EL} is correctly received,

$$MV_e^{BL} = \lfloor \alpha(\frac{1}{2}MV_e^{EL} + \frac{1}{2}MV_{Pred}^{EL}) + (1-\alpha)MV_{Med}^{BL} \rceil, \tag{4.1}$$

$$MV_{Pred}^{EL} = \sum_i w(i) * MV_o^{EL}(i) / \sum_i w(i), \tag{4.2}$$

$$MV_{Med}^{BL} = Median\{MV_{MVE}^{BL} \bigcup MV_{Nb}^{BL}(k)\}. \tag{4.3}$$

BL Case 2: If MV_e^{BL} and MV_e^{EL} are both lost,

$$MV_e^{BL} = MV_{Med}^{BL}. \tag{4.4}$$

Enhancement Layer (EL): The lost motion vector $MV_{o/e}^{EL}$ in the EL current frame f_c^{EL} can be reconstructed in two cases, where $MV_{o/e}^{EL}$ denotes either MV_o^{EL} or MV_e^{EL} for a specific odd or even frame number. In case 1, if $MV_{o/e}^{EL}$ is lost but MV_e^{BL} is correctly received, $MV_{o/e}^{EL}$ can be recovered by adaptively averaging over two synthetic motion vectors. One is an approximate motion vector $2 * \frac{1}{2} MV_e^{BL} = MV_e^{BL}$ in BL corresponding frame f_c^{BL}. The other is a median motion vector $MV_{o/e,Med}^{EL}$ based on the MVE estimated and s neighbors' motion vectors in the same EL odd/even frame. In case 2, if MV_e^{BL} and $MV_{o/e}^{EL}$ are both lost, $MV_{o/e}^{EL}$ can be recovered using the median motion vector $MV_{o/e,Med}^{EL}$ in the same EL odd/even frame.

EL Case 1: If $MV_{o/e}^{EL}$ is lost but MV_e^{BL} is correctly received,

$$MV_{o/e}^{EL} = \lfloor \alpha MV_e^{BL} + (1-\alpha) MV_{o/e,Med}^{EL} \rceil, \tag{4.5}$$

$$MV_{o/e,Med}^{EL} = Median\{MV_{MVE}^{EL} \bigcup MV_{o/e,Nb}^{EL}(k)\}. \tag{4.6}$$

EL Case 2: If $MV_{o/e}^{EL}$ and MV_e^{BL} are both lost,

$$MV_{o/e}^{EL} = MV_{o/e,Med}^{EL}. \tag{4.7}$$

Slice Interleaving

In order to improve the performance against burst packet losses and reduce error propagation across multiple frames, a new slice interleaving approach is developed to make the average distance between two contiguous slices as far as possible. The slice tool can be used at the encoder to generate independently decodable slices with the cost of some loss in coding efficiency. The main idea of this approach is to rearrange the slices according to a predefined interleaving structure, which would be designed in such a way that the contiguous slices are distributed as far as possible. In [144], an optimum ordering technique was developed for dispersed-dot ordered dithering for halftone image processing. The method was used for obtaining the optimum index for adding dots to lattices. The optimum index matrix is a square matrix and devised with a simple rule: First, fill each cell of the matrix with a successive integer (e.g.

starting from 1 in raster scanning order). Second, reorder them such that the average distance between two successive numbers is as far as possible in the matrix. It can be rotated or mirrored without affecting the property of maximizing average distance. The optimum index matrix can be defined recursively and three concrete examples are illustrated in Figure 4.2.

$$A_2 = \begin{bmatrix} 1 & 4 \\ 3 & 2 \end{bmatrix}, A_{2n} = \begin{bmatrix} 4 \times A_n - 3 & 4 \times A_n \\ 4 \times A_n - 1 & 4 \times A_n - 2 \end{bmatrix}. \tag{4.8}$$

$$\begin{bmatrix} 1 & 4 \\ 3 & 2 \end{bmatrix}$$

(a)

$$\begin{bmatrix} 1 & 13 & 4 & 16 \\ 9 & 5 & 12 & 8 \\ 3 & 15 & 2 & 14 \\ 11 & 7 & 10 & 6 \end{bmatrix}$$

(b)

$$\begin{bmatrix} 1 & 49 & 13 & 61 & 4 & 52 & 16 & 64 \\ 33 & 17 & 45 & 29 & 36 & 20 & 48 & 32 \\ 9 & 57 & 5 & 53 & 12 & 60 & 8 & 56 \\ 41 & 25 & 37 & 21 & 44 & 28 & 40 & 24 \\ 3 & 51 & 15 & 63 & 2 & 50 & 14 & 62 \\ 35 & 19 & 47 & 31 & 34 & 18 & 46 & 30 \\ 11 & 59 & 7 & 55 & 10 & 58 & 6 & 54 \\ 43 & 27 & 39 & 23 & 42 & 26 & 38 & 22 \end{bmatrix}$$

(c)

Fig. 4.2. Optimum index matrixes of different size.

We propose a new slice interleaving approach for a set of contiguous slices in a group of pictures (GOP) using the optimum ordering technique described above. The number of contiguous slices in one frame is designed to be equal to the number of consecutive frames in a GOP, hence a set of contiguous slices in a GOP can be represented by a square matrix. Figure 4.3 illustrates an example of slice interleaving with an 8x8 optimum index matrix for a set of 64 contiguous slices among 8 consecutive frames in a GOP. The frame numbers of a group of consecutive frames are denoted in a temporally ascending order along the horizontal axis. The slice numbers of a set of contiguous slices, which are labeled by successive integers, are denoted in a spatially ascending order along the vertical axis. Contiguous slices in one frame are rearranged into disjoint positions by maximizing the average distance between each other. Each

frame consists of a few independently decodable slices and each slice is encapsuled in a network abstraction layer unit (NALU). Slice interleaving is performed by a square interleaver on a set of NALUs containing contiguous slices after initial placement. Each NALU containing a single slice is interleaved according to the optimum index matrix and then packetized in raster scan order.

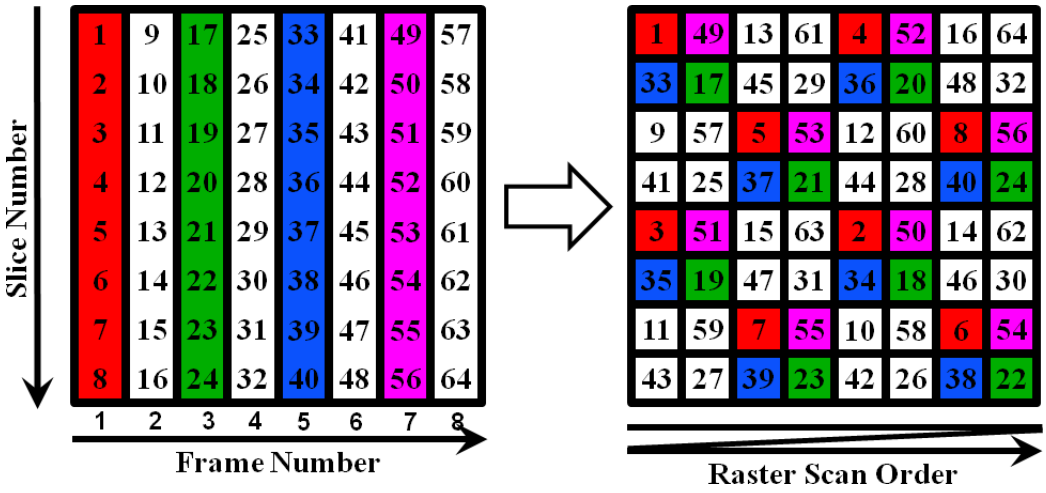


Fig. 4.3. The proposed slice interleaving scheme in a GOP (8 frames).

Similarly, flexible macroblock ordering (FMO) is capable of distributing adjoining macroblock errors to the entire frame as equally as possible to avoid error accumulation in a certain region. The FMO tool can be used at the encoder to assign each macroblock of a frame to a certain slice group by a macroblock allocation map (MBAmap), which requires additional computation and causes some loss in coding efficiency. For further comparison, FMO Type 1 [132] is used to generate two slice groups and each one contains four independently dispersed slices (totally 8 slices per frame), which is complied with our proposed slice interleaving approach.

4.2 System Implementation

A two-layer spatial-temporal scalable video coding (SVC) system was developed based on JSVM 9.8 [145], which was the last version officially supporting error concealment tools. Sixty-four independently decodable slices in a GOP (8 frames in BL and EL separately) are interleaved using the 8x8 optimum index matrix in Figure 4.3 at the encoder and correspondingly de-interleaved at the decoder. Each interleaved slice was encapsulated into a single NALU during packetization. The parameters and syntax elements of the 8x8 optimum index matrix and FMO Type 1 are independently encoded into a picture parameter set (PPS) and transmitted to the decoder. The extra bits used for encoding these information were counted in the total bitrates. We modified the JSVM reference decoder to deal with lost NALUs and conceal damaged frames. It is able to manage the order of the NALUs received at decoder and identify the decoding information of the layer and slice in the received NALUs. Similar to [146], a block-based status map is developed for each layer to inform the decoder to decode available blocks and conceal lost blocks. The status map is reinitialized at the beginning of decoding each slice.

Burst packet losses were simulated by removing NALUs from the encoded SVC bitstreams based on random burst packet losses at different rates. Gilbert's two-state Markov model [147, 148] was used to independently generate random burst packet loss patterns. This model can reasonably approximate Internet transmission [17]. In the good state G, all packets are correctly received, while in the bad state B, all packets are lost. Two transition probabilities, p_{GB} for going from G to B and p_{BG} for going from B to G, are sufficient to define the model. Moreover, other two quantities are preferred to use: average burst packet loss probability $P_B = Pr(B) = p_{GB}/(p_{GB} + p_{BG})$, the same as the well-defined burst packet loss rate (BPLR), and average burst packet loss length $L_B = 1/p_{BG}$. The network simulation parameters are defined using a pair (P_B, L_B). In our experiments, SVC video transmission over burst packet loss channels are simulated in two scenarios: first corresponding to

$(P_B, L_B) = (10\%, 5)$ and second corresponding to $(P_B, L_B) = (20\%, 4)$. Damaged frame are recovered by different error concealment methods, including three proposed inter-layer methods using the two new approaches and FMO described above, i.e. (1) motion vector averaging (MVAvg), (2) motion vector averaging and slice interleaving (MVAvg+SliceIntlv) and (3) motion vector averaging, slice interleaving and FMO (MVAvg+SliceIntlv+FMO), and two existing methods, i.e. (4) motion copy (MC) [18] and (5) motion projection (MP) [142]. Because the original MC and MP methods are developed for single-layer error concealment, the BLSkip-based extensions [18] of the two methods are developed for inter-layer EC.

4.3 Experimental Results

Three video sequences with 300 frames, i.e. *Bus*, *Football* and *Foreman*, were used to test our SVC system. The *Bus* sequence contains slow and homogenous motion while the *Football* sequence has fast and chaotic motion. The *Foreman* sequence involves normal motion and scene changes. Our experiments used the same quantization parameter (QP) for encoding BL and EL frames (QP=28, 32, 36 and 40) to evaluate different error concealment methods, running on a Linux desktop with a 2.8 GHz Quad-core CPU and 4 GB RAM. The frame coding structure was "IPP$\cdots$" with I-frame refresh after 2 successive P-frame GOPs in BL and 4 successive P-frame GOPs in EL. For low complexity, we constantly set the parameter $\alpha = 0.5$ and employed the 4-neighborhood of adjoining blocks $s = 4$. The average PSNR value of the Y component (Y-PSNR) of damaged frames was used as an objective visual quality measurement. The Y-PSNR was obtained by averaging the results of 50 random burst packet loss patterns at each BPLR to ensure statistical significance of the results. Each burst packet loss pattern has 20 temporally circular shifts across the entire frames, in total 20×50=1000 realizations of burst packet losses at a BPLR.

Table 4.1 demonstrates the average decoding time per frame of the existing and proposed error concealment methods. The results show that the computational time of the proposed MVAvg method is slightly longer than the MP and MC method. It can be observed that the decoding delay (time difference between MVAvg-based methods) caused by slice de-interleaving is relatively shorter than that introduced by FMO.

The Y-PSNR of the first 60 frames of BL and EL *Football* sequence are illustrated in Figure 4.4 and 4.5 with two concrete burst packet loss patterns at BPLR 10% and 20%, where the vertical dash lines indicate the damaged frames where burst packet losses occurred. The visual quality in the enhancement layer is recovered slightly faster than that in the base layer and the Y-PSNR drop in the enhancement layer is comparatively smaller than that in the base layer. Visual distortion due to poorly

Table 4.1
Average Decoding Time per Frame of the Existing and Proposed Error Concealment Methods (in Milliseconds)

	Bus	*Football*	*Foreman*
Motion Copy (MC)	16.36	16.59	16.25
Motion Projection (MP)	17.33	17.82	17.14
MVAvg	17.94	18.53	17.68
MVAvg+SliceIntlv	19.95	20.60	19.72
MVAvg+SliceIntlv+FMO	24.08	24.76	23.87

concealed motion vectors is barely observed in the proposed three methods, except for fast moving objects.

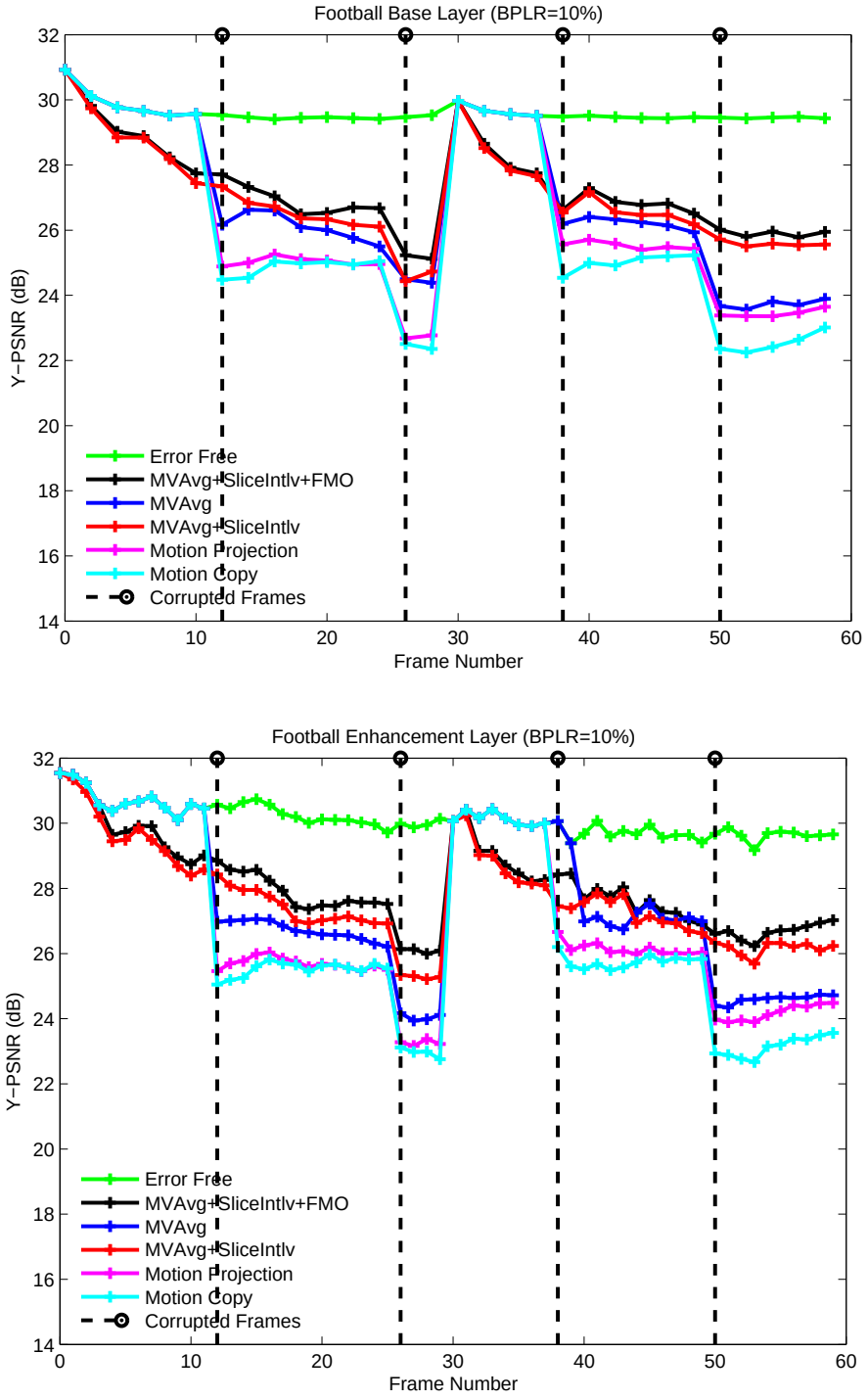


Fig. 4.4. Y-PSNR of the *Football* BL & EL frames (BPLR=10%).

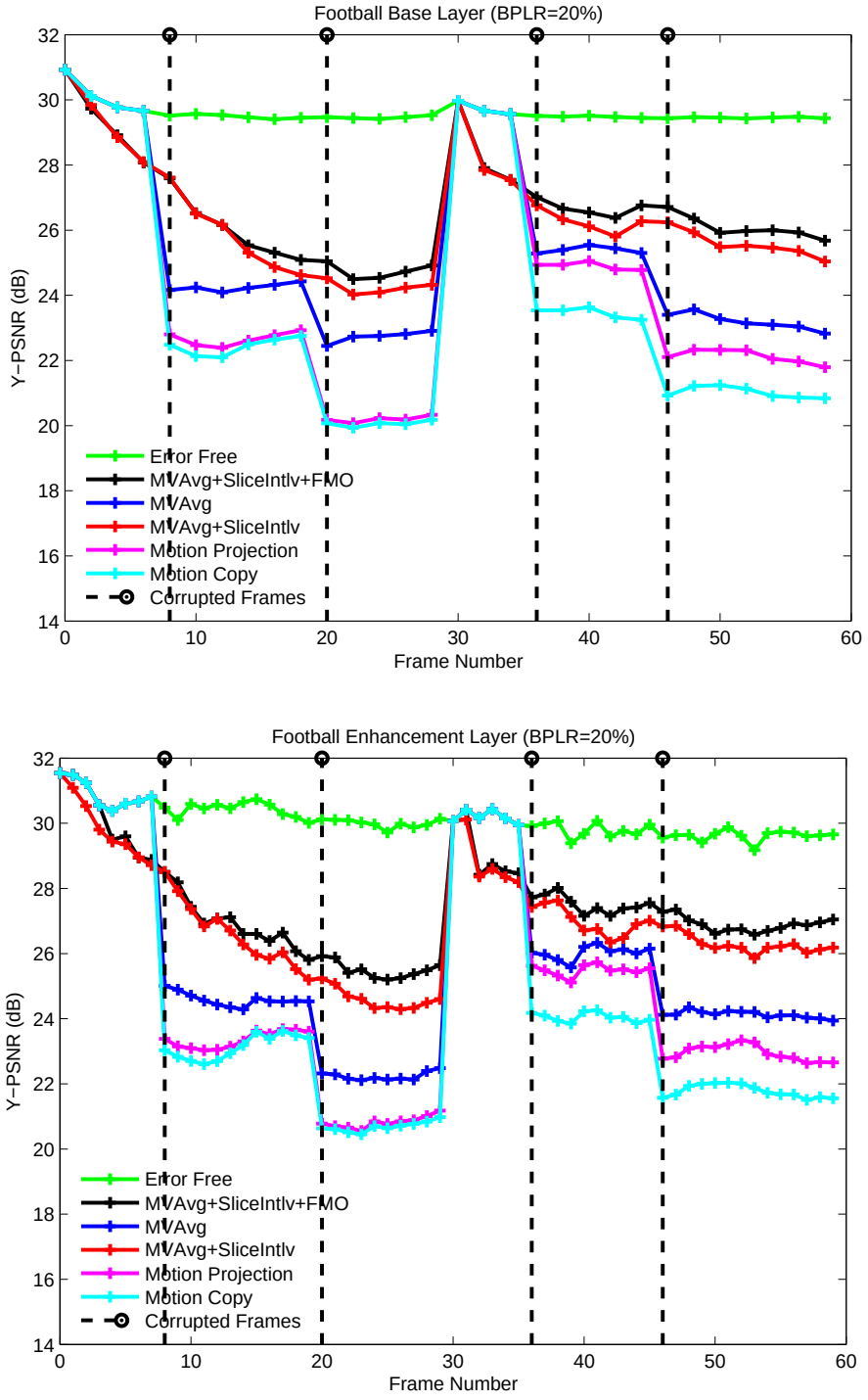


Fig. 4.5. Y-PSNR of the *Football* BL & EL frames (BPLR=20%).

Compared with the perfect reconstruction from the error-free channel, the operational rate-distortion (RD) plots for various error concealment methods are shown in Figure 4.6, 4.7 and 4.8. The proposed MVAvg method based on motion vector averaging is more effective than two existing methods in reducing the visual quality degradation caused by burst packet losses and error propagation. The RD plots illustrate that the Y-PSNR of the proposed MVAvg method is 0.9dB-3.2dB higher than the existing MC method and is 0.3dB-2.1dB higher than the existing MP method. The proposed MVAvg+SliceIntlv+FMO and MVAvg+SliceIntlv methods outperform the other methods in significantly improving the visual quality. In fact, the RD plots of the two methods are very close at low bitrate and low BPLR. The MVAvg+SliceIntlv+FMO method is superior to the MVAvg+SliceIntlv method only at high bitrate and high BPLR, because additional bit overhead for encoding FMO information undermines the coding efficiency at low bitrate. Therefore, considering the tradeoff between complexity and performance, The the proposed MVAvg+SliceIntlv method is more suitable for low burst packet loss channel.

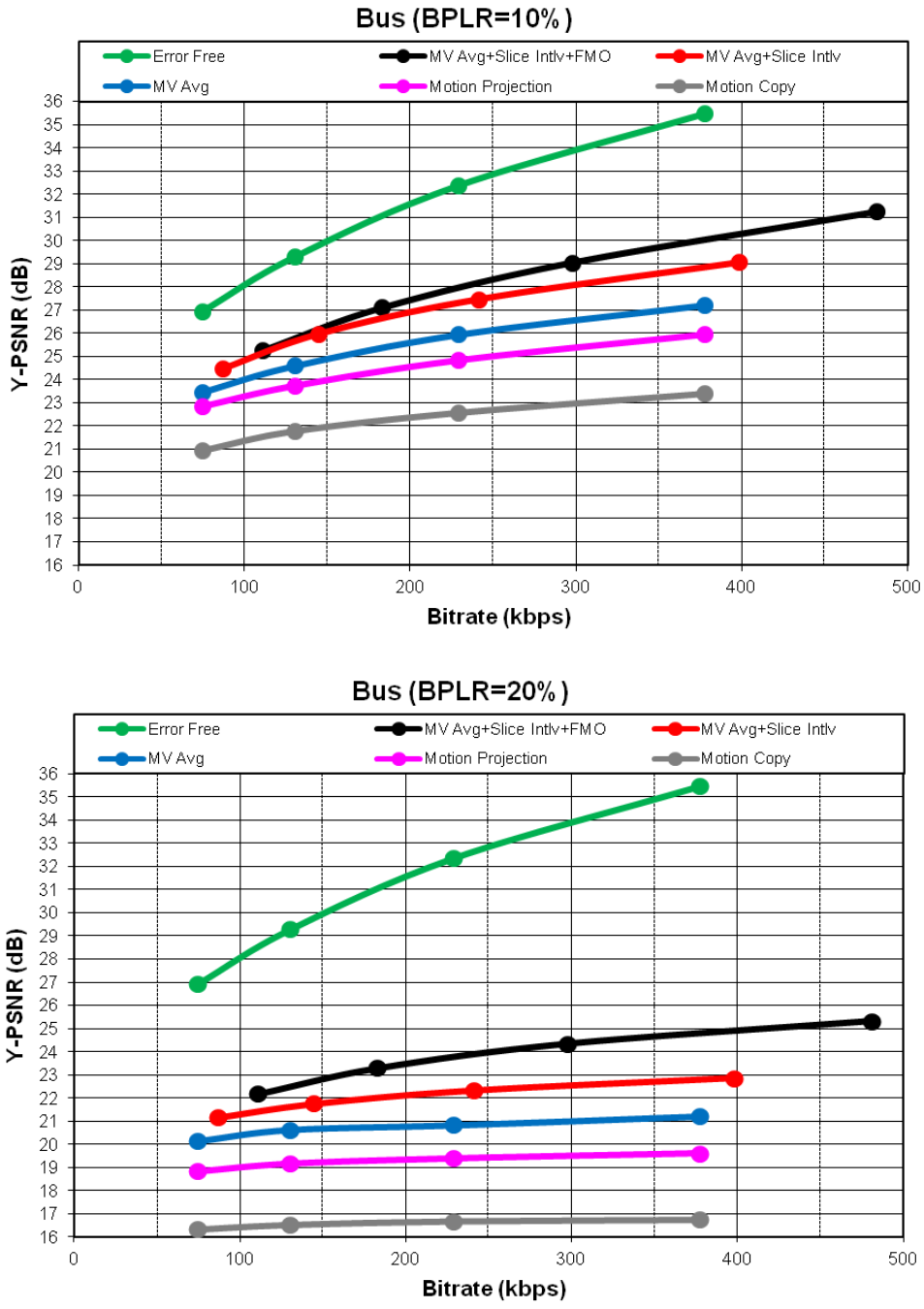


Fig. 4.6. Rate-Distortion of the *Bus* sequence (BPLR=10%, 20%).

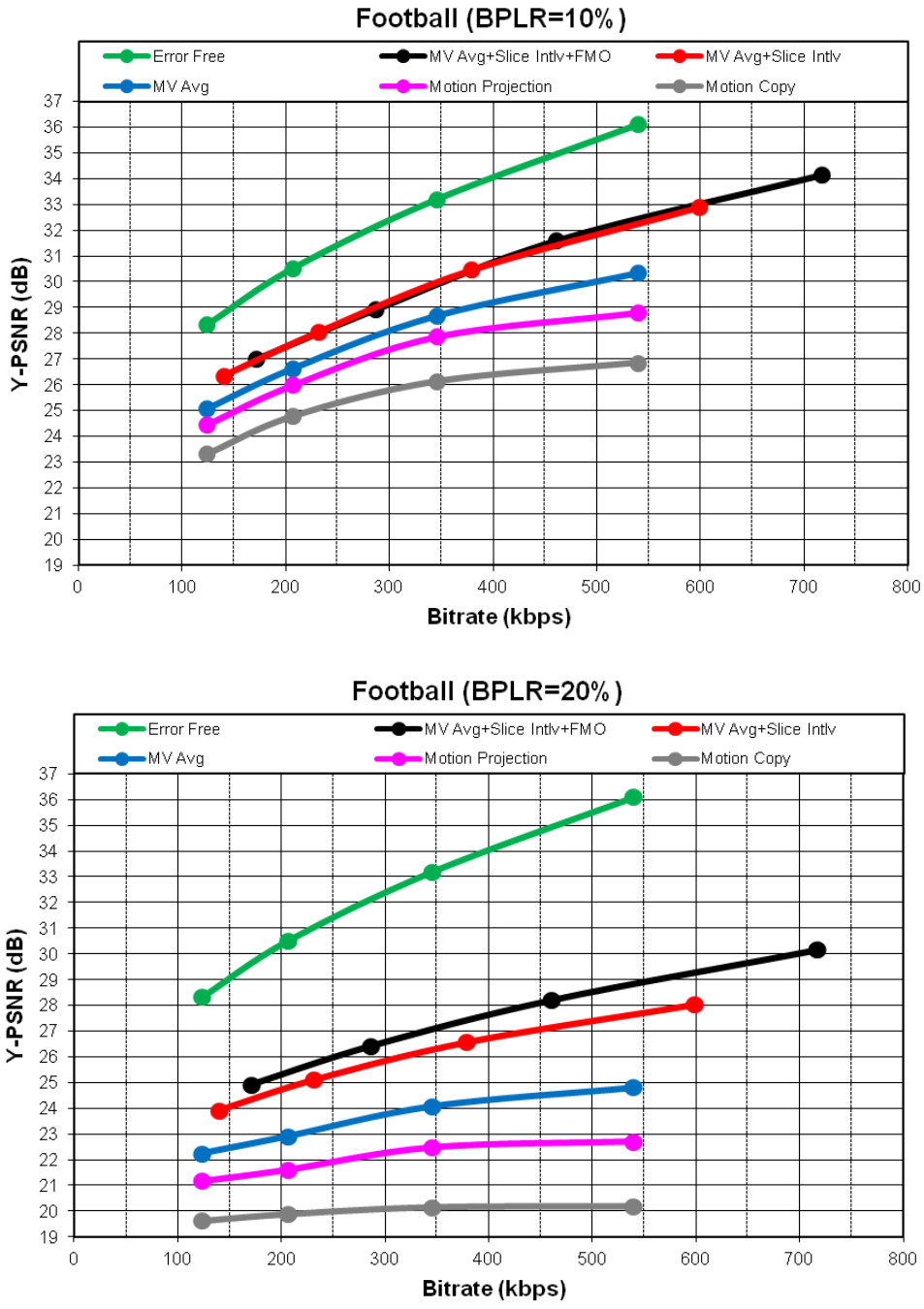


Fig. 4.7. Rate-Distortion of the *Football* sequence (BPLR=10%, 20%).

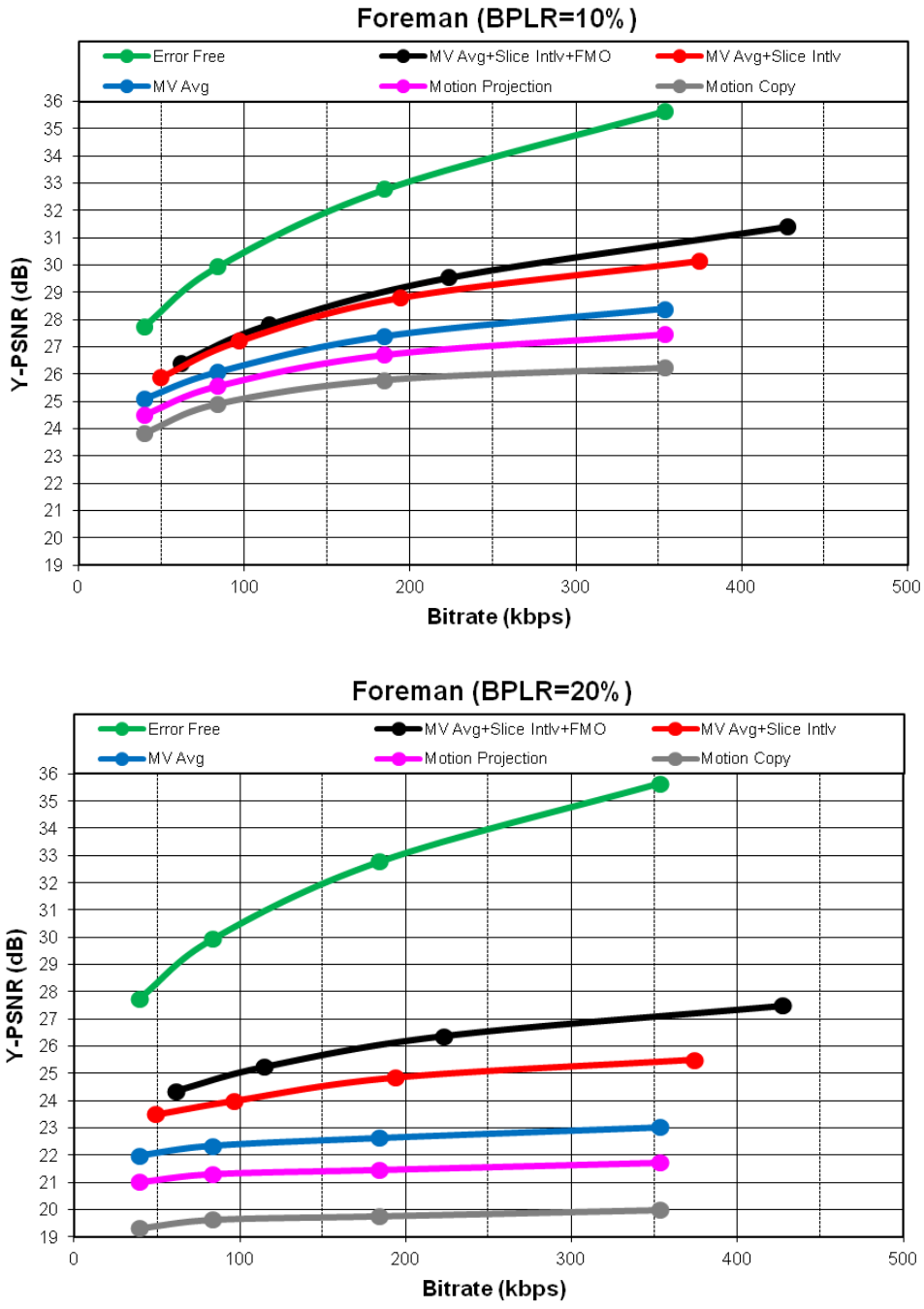


Fig. 4.8. Rate-Distortion of the *Foreman* sequence (BPLR=10%, 20%).

5. CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

In this thesis we describe several visual saliency models in the frequency domain in Chapter 2, a hazmat sign image analysis system (MERGE) using visual saliency for location detection and content recognition in Chapter 3, and several error concealment methods for scalable video coding (SVC) in chapter 4.

For visual saliency models in the frequency domain, we develop separate and composite visual saliency model families for frequency domain visual saliency models. We propose six visual saliency models based on new spectrum processing methods and an entropy-based saliency map selection approach. We propose an entropy-based saliency map selection approach to select a "good" final saliency map among the set of map candidates. A group of extended saliency models that extends each proposed visual saliency models are also developed by incorporating both separate and composite model families and using variant color spaces. Experimental results show that the six best extended models are more accurate and efficient than most state-of-the-art models in predicting eye fixation on standard image datasets.

For hazmat sign image analysis system (MERGE), we develop hazmat sign location detection and content recognition methods based on visual saliency. We use the one of our proposed frequency domain models to extract salient regions that are likely to contain hazmat sign candidates and then use a Fourier descriptor based contour matching method to locate the border of hazmat signs in these regions. This visual saliency based approach is able to increase the accuracy of sign location detection, significantly reduce the number of false positives, and speed up the image analysis process. This approach improves the accuracy of existing methods presented in [14, 15]. We also propose a color recognition method to interpret the color inside

the detected hazmat signs. Our three image datasets consists of images taken in the working field and outdoor field under variant lighting and weather conditions, distances, and perspectives.

For error concealment for scalable video coding (SVC), we develop two error concealment approaches robust to burst packet losses, i.e. inter-layer motion vector averaging and slice interleaving using optimum ordering. A two-layer spatial-temporal scalable video coding system are decribed to evaluate the existing and proposed error concealment methods. Experimental results confirmed that the proposed error concealment methods outperform two existing methods in reducing the impact of burst packet losses and error propagation.

The main contributions of visual saliency models in the frequency domain are:

- We investigate bottom-up visual saliency using spectral analysis approaches.
- We develop separate and composite visual saliency model families for frequency domain models.
- We propose six visual saliency models based on different spectrum processing.
- We propose an entropy-based saliency map selection approach.
- We develop an evaluation tool for benchmarking visual saliency models.

The main contributions of image analysis system for hazmat sign detection and recognition are:

- We develop a hazmat sign location detection and content recognition system using visual saliency.
- We used one of our proposed frequency domain models to extract salient regions.
- We developed a Fourier descriptor based contour matching method to locate the border of hazmat signs.

- We proposed a color recognition method to interpret the color inside the detected hazmat signs.
- We collected three hazmat sign image datasets.

The main contributions of error concealment methods for SVC are:

- We investigated the impact of burst packet loss and error propagation in base and enhancement layers.
- We explored inter-layer spatial and temporal correlations for error concealment against burst packet loss.
- We proposed two error concealment methods to enhance error recovery and visual quality:
- (1) Inter-layer motion vector averaging
- (2) Slice interleaving using optimum ordering
- We developed a two-layer spatial-temporal scalable video coding system for evaluation.

5.2 Future Work

Our long term goal for MERGE is to develop a hazmat sign image analysis system capable of automatically recognizing hazmat signs from images acquired up to 300 feet and providing real-time guide information to first responders to identify the hazardous materials and determine what specialty equipment, procedures and precautions should be taken in the event of an emergency.

One problem is the overall accuracy of our hazmat sign image analysis methods. The accuracy needs to be improved. This can be done by improving our current sign location detection approach and developing more robust color recognition techniques. We may be able to use supper-resolution image reconstruction methods [126, 127] to

refine the hazmat sign images. It can improve the location detection accuracy at even longer distances and it is more useful for blurred and low resolution hazmat signs. We can also employ proper color calibration [51]. This can help the color recognition technique to recognize colored hazmat signs more accurately. One could also use character recognition methods to interpret the text inside the detected hazmat signs when the image resolution is relatively high.

For visual saliency models in the frequency domain, one direction of future work is testing our proposed visual saliency models using more eye fixation image datasets. One could also study the tradeoff between accuracy and speed of the proposed frequency domain saliency models for practical applications. Another direction is combining several saliency models to achieve better accuracy of predicting eye fixation and hazmat sign image analysis.

For error concealment for scalable video coding (SVC), one direction of future work is testing our proposed error concealment models on high resolution video sequences.

5.3 Publications Resulting from This Work

Conference Papers

1. **Bin Zhao** and Edward J. Delp, "Visual Saliency Models Based on Spectrum Processing," *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, Waikoloa Beach, HI, USA, January 2015. (Accepted)

2. **Bin Zhao**, Albert Parra, and Edward J. Delp, "Mobile-Based Hazmat Sign Detection and Recognition," *Proceedings of the IEEE Global Conference on Signal and Information Processing*, no. 6736996, pp. 735-738, Austin, TX, USA, December 2013. (Invited Paper)

3. **Bin Zhao** and Edward J. Delp, "Inter-layer Error Concealment for Scalable Video Coding," *Proceedings of the IEEE International Conference on Multimedia and Expo*, no. 6607539, pp. 1-6, San Jose, CA, USA, July 2013.

4. **Bin Zhao**, "Interleaving-Based Error Concealment for Scalable Video Coding System," *Proceedings of the IEEE Visual Communications and Image Processing Conference*, no. 6115965, pp. 1-4, Tainan City, Taiwan, November 2011.

5. Albert Parra, **Bin Zhao**, Joonsoo Kim, and Edward J. Delp, "Recognition, Segmentation and Retrieval of Gang Graffiti Images on a Mobile Device," *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, no. 6698996, pp. 178-183, Waltham, MA, USA, November 2013.

6. Albert Parra, **Bin Zhao**, Andrew Haddad, Mireille Boutin, and Edward J. Delp, "Hazardous Material Sign Detection and Recognition," *Proceedings of the IEEE International Conference on Image Processing*, no. 6738544, pp. 2640-2644, Melbourne, Australia, September 2013.

Journal Papers

1. **Bin Zhao** and Edward J. Delp, "Biologically-Inspired Visual Saliency Models Using Spectrum Processing," in preparation.

2. **Bin Zhao**, Albert Parra, and Edward J. Delp, "Hazmat Sign Detection and Recognition Using Visual Saliency," in preparation.

LIST OF REFERENCES

LIST OF REFERENCES

[1] ERG, available: http://www.phmsa.dot.gov/hazmat/library/erg.

[2] L. Itti, "Models of bottom-up and top-down visual attention," Ph.D. dissertation, California Institute of Technology, 2000.

[3] C. Koch and S. Ullman, "Shifts in selective visual attention: Towards the underlying neural circuitry," *Human Neurobiology*, vol. 4, no. 4, pp. 219–227, April 1985.

[4] A. M. Treisman and G. Gelade, "A feature-integration theory of attention," *Cognitive Psychology*, vol. 12, no. 1, pp. 97–136, January 1980.

[5] J. M. Wolfe, "Guided search 2.0 A revised model of visual search," *Psychonomic Bulletin & Review*, vol. 1, no. 2, pp. 202–238, 1994.

[6] J. M. Wolfe and T. S. Horowitz, "What attributes guide the deployment of visual attention and how do they do it?" *Nature Reviews Neuroscience*, vol. 5, no. 6, pp. 495–501, June 2004.

[7] S. Frintrop, E. Rome, and H. I. Christensen, "Computational visual attention systems and their cognitive foundations: A survey," *ACM Transactions on Applied Perception*, vol. 7, no. 1, pp. 6:1–6:39, January 2010.

[8] A. Toet, "Computational versus psychophysical bottom-up image saliency: A comparative evaluation study," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 11, pp. 2131–2146, November 2011.

[9] A. Borji and L. Itti, "State-of-the-art in visual attention modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 185–207, January 2013.

[10] A. Borji, D. N. Sihite, and L. Itti, "Quantitative analysis of human-model agreement in visual saliency modeling: A comparative study," *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 55–69, January 2013.

[11] United States Department of Transportation, *Code of Federal Regulations, Title 49 - Transportation*, 2012nd ed., October 2012.

[12] WISER, available: http://wiser.nlm.nih.gov.

[13] MERGE, available: http://www.hazmat-signs.org.

[14] B. Zhao, A. Parra, and E. J. Delp, "Mobile-based hazmat sign detection system," *Proceedings of the IEEE Global Conference on Signal and Information Processing*, pp. 735–738, December 2013, Austin, TX, USA.

[15] A. Parra, B. Zhao, A. Haddad, M. Boutin, and E. J. Delp, "Hazardous material sign detection and recognition," *Proceedings of the IEEE International Conference on Image Processing*, pp. 2640–2644, September 2013, Melbourne, Australia.

[16] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, September 2007.

[17] K. Stuhlmüller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 1012–1032, June 2000.

[18] Y. Guo, Y. Chen, Y.-K. Wang, H. Li, M. M. Hannuksela, and M. Gabbouj, "Error resilient coding and error concealment in scalable video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 6, pp. 781–795, June 2009.

[19] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: An overview," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 4, pp. 1103–1127, November 2000.

[20] L. Itti, "Automatic foveation for video compression using a neurobiological model of visual attention," *IEEE Transactions on Image Processing*, vol. 13, no. 10, pp. 1304–1318, October 2004.

[21] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," *ACM Transactions on Graphics*, vol. 26, no. 3, pp. 1–9, July 2007.

[22] J. Han, K. N. Ngan, M. Li, and H. Zhang, "Unsupervised extraction of visual attention objects in color images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 1, pp. 141–145, January 2006.

[23] D. Walther and C. Koch, "Modeling attention to salient proto-objects," *Neural Networks*, vol. 19, no. 9, pp. 1395–1407, November 2006.

[24] A. Mogelmose, M. M. Trivedi, and T. B. Moeslund, "Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1484–1497, December 2012.

[25] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, November 1998.

[26] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," *Proceedings of the Annual Conference on Neural Information Processing Systems*, pp. 545–552, December 2006, Vancouver, BC, Canada.

[27] N. D. B. Bruce and J. K. Tsotsos, "Saliency, attention, and visual search: An information theoretic approach," *Journal of Vision*, vol. 9, no. 3, pp. 1–24, March 2009.

[28] X. Hou and L. Zhang, "Dynamic visual attention: Searching for coding length increments," *Proceedings of the Annual Conference on Neural Information Processing Systems*, pp. 681–688, December 2008, Vancouver, BC, Canada.

[29] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1597–1604, June 2009, Miami, FL, USA.

[30] C. Guo, Q. Ma, and L. Zhang, "Spatio-temporal saliency detection using phase spectrum of quaternion Fourier transform," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2008, Anchorage, AK, USA.

[31] C. Guo and L. Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 185–198, January 2010.

[32] P. Bian and L. Zhang, "Visual saliency: A biologically plausible contourlet-like frequency domain approach," *Cognitive Neurodynamics*, vol. 4, no. 3, pp. 189–198, September 2010.

[33] X. Hou, J. Harel, and C. Koch, "Image signature: Highlighting sparse salient regions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 194–201, January 2012.

[34] B. Schauerte and R. Stiefelhagen, "Predicting human gaze using quaternion DCT image signature saliency and face detection," *Proceedings of the IEEE Workshop on Applications of Computer Vision*, pp. 137–144, January 2012, Breckenridge, CO, USA.

[35] J. Li, M. D. Levine, X. An, X. Xu, and H. He, "Visual saliency based on scale-space analysis in the frequency domain," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 4, pp. 996–1010, April 2013.

[36] T. A. Ell, "Quaternion-Fourier transforms for analysis of two-dimensional linear time-invariant partial differential systems," *Proceedings of the IEEE Conference on Decision and Control*, vol. 2, pp. 1830–1841, December 1993, San Antonio, TX, USA.

[37] T. A. Ell and S. J. Sangwine, "Hypercomplex Fourier transforms of color images," *IEEE Transactions on Image Processing*, vol. 16, no. 1, pp. 22–35, January 2007.

[38] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell, "SUN: A bayesian framework for saliency using natural statistics," *Journal of Vision*, vol. 8, no. 7, pp. 1–20, December 2008.

[39] Y. Fang, W. Lin, B.-S. Lee, C.-T. Lau, Z. Chen, and C.-W. Lin, "Bottom-up saliency detection model based on human visual sensitivity and amplitude spectrum," *IEEE Transactions on Multimedia*, vol. 14, no. 1, pp. 187–198, February 2012.

[40] Z. Li, "A saliency map in primary visual cortex," *Trends in Cognitive Sciences*, vol. 6, no. 1, pp. 9–16, January 2002.

[41] Z. Li and P. Dayan, "Pre-attentive visual selection," *Neural Networks*, vol. 19, no. 9, pp. 1437–1439, November 2006.

[42] C. A. Curcio, K. R. Sloan, R. E. Kalina, and A. E. Hendrickson, "Human photoreceptor topography," *The Journal of Comparative Neurology*, vol. 292, no. 4, pp. 497–523, February 1990.

[43] E. D. Montag, *Rods and Cones*, available: http://www.cis.rit.edu/people/faculty/montag/vandplite/pages/chap_9/ch9p1.html.

[44] C. Blakemore and F. W. Campbell, "On the existence of neurones in the human visual system selectively sensitive to the orientation and size of retinal images," *The Journal of Physiology*, vol. 203, no. 1, pp. 237–260, July 1969.

[45] D. Casasent and D. Psaltis, "New optical transforms for pattern recognition," *Proceedings of the IEEE*, vol. 65, no. 1, pp. 77–84, January 1977.

[46] P. Cavanagh, "Size and position invariance in the visual system," *Perception*, vol. 7, no. 2, pp. 167–177, 1978.

[47] ——, "Size invariance: Reply to Schwartz," *Perception*, vol. 10, no. 4, pp. 469–474, 1981.

[48] L. O. Harvey and V. V. Doan, "Visual masking at different polar angles in the two-dimensional Fourier plane," *Journal of the Optical Society of America A*, vol. 7, no. 1, pp. 116–127, January 1990.

[49] E. L. Schwartz, "Spatial mapping in the primate sensory projection: Analytic structure and relevance to perception," *Biological Cybernetics*, vol. 25, no. 4, pp. 181–194, 1977.

[50] ——, "Computational anatomy and functional architecture of striate cortex: A spatial mapping approach to perceptual coding," *Vision Research*, vol. 20, no. 8, pp. 645–669, 1980.

[51] H.-C. Lee, *Color Imaging Science*. Cambridge, UK: Cambridge University Press, 2005.

[52] M. D. Fairchild, *Color Appearance Models*. Chichester, UK: Wiley-IS&T, 2013.

[53] S. Süsstrunk, R. Buckley, and S. Swen, "Standard RGB color spaces," *Proceedings of the Color Imaging Conference: Color Science, Systems, and Applications*, pp. 127–134, November 1999, Scottsdale, AZ, USA.

[54] S. Engel, X. Zhang, and B. Wandell, "Colour tuning in human visual cortex measured with functional magnetic resonance imaging," *Nature*, vol. 388, no. 6637, pp. 68–71, July 1997.

[55] W. R. Hamilton, *Elements of Quaternions*. Dublin, Ireland: The University of Dublin Press, 1866.

[56] P. Bian and L. Zhang, "Biological plausibility of spectral domain approach for spatiotemporal visual saliency," *Proceedings of the International Conference Neuro-Information Processing*, vol. 5506, pp. 251–258, November 2008, Auckland, New Zealand.

[57] B. Schauerte and R. Stiefelhagen, "Quaternion-based spectral saliency detection for eye fixation prediction," *Proceedings of the European Conference on Computer Vision*, pp. 116–129, October 2012, Florence, Italy.

[58] W. Feng and B. Hu, "Quaternion discrete cosine transform and its application in color template matching," *Proceedings of the International Congress on Image and Signal Processing*, vol. 2, pp. 252–256, May 2008, Sanya, China.

[59] K. Castleman, *Digital Image Processing*. New York, USA: Prentice-Hall, 1996.

[60] A. V. Oppenheim and J. S. Lim, "The importance of phase in signals," *Proceedings of the IEEE*, vol. 69, no. 5, pp. 529–541, May 1981.

[61] M. H. Hayes, J. S. Lim, and A. V. Oppenheim, "Signal reconstruction from phase or magnitude," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, no. 6, pp. 672–680, December 1980.

[62] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," *Mathematics of Computation*, vol. 19, no. 90, pp. 297–301, April 1965.

[63] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, June 1996.

[64] D. J. Graham and D. J. Field, "Natural images: Coding efficiency," *Encyclopedia of Neuroscience*, pp. 19–27, 2009.

[65] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2007, Minneapolis, MN, USA.

[66] C. A. Poynton, "Rehabilitation of gamma," *Proceedings of the SPIE Conference on Human Vision and Electronic Imaging III*, vol. 3299, pp. 232–249, January 1998, San Jose, CA, USA.

[67] S. S. Stevens, "On the psychophysical law," *Psychological Review*, vol. 64, no. 3, pp. 153–181, May 1957.

[68] R. J. Peters, A. Iyer, L. Itti, and C. Koch, "Components of bottom-up gaze allocation in natural images," *Vision Research*, vol. 45, no. 18, pp. 2397–2416, August 2005.

[69] S. Kullback, *Information Theory and Statistics*. New York, USA: Wiley, 1959.

[70] L. Itti and P. Baldi, "A principled approach to detecting surprising events in video," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 631–637, June 2005, San Diego, CA, USA.

[71] T. Jost, N. Ouerhani, R. von Wartburg, R. Mri, and H. Hgli, "Assessing the contribution of color in visual attention," *Computer Vision and Image Understanding*, vol. 100, no. 1-2, pp. 107–123, Oct.-Nov. 2005.

[72] D. M. Green and J. A. Swets, *Signal Detection Theory and Psychophysics*. New York, USA: Wiley, 1966.

[73] B. W. Tatler, R. J. Baddeley, and I. D. Gilchrist, "Visual correlates of fixation selection: Effects of scale and time," *Vision Research*, vol. 45, no. 5, pp. 643–659, March 2005.

[74] F. Larsson, M. Felsberg, and P.-E. Forssén, "Correlating fourier descriptors of local patches for road sign recognition," *IET Computer Vision*, vol. 5, no. 4, pp. 244–254, July 2011.

[75] T. Gevers and A. W. M. Smeulders, "Color-based object recognition," *Pattern Recognition*, vol. 32, no. 3, pp. 453–464, March 1999.

[76] C. Grigorescu and N. Petkov, "Distance sets for shape filters and shape recognition," *IEEE Transactions on Image Processing*, vol. 12, no. 10, pp. 1274–1286, October 2003.

[77] D. Gossow, J. Pellenz, and D. Paulus, "Danger sign detection using color histograms and SURF matching," *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics*, pp. 13–18, October 2008, Sendai, Japan.

[78] A. Broggi, P. Cerri, P. Medici, P. Porta, and G. Ghisio, "Real time road signs recognition," *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 981–986, June 2007, Istambul, Turkey.

[79] K. L. Bouman, G. Abdollahian, M. Boutin, and E. J. Delp, "A low complexity sign detection and text localization method for mobile applications," *IEEE Transactions on Multimedia*, vol. 13, no. 5, pp. 922–934, October 2011.

[80] L. Song and Z. Liu, "Color-based traffic sign detection," *Proceedings of the International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering*, pp. 353–357, June 2012, Chengdu, China.

[81] G. Loy and N. Barnes, "Fast shape-based road sign detection for a driver assistance system," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, pp. 70–75, September 2004, Stockholm, Sweden.

[82] S. Maldonado-Bascon, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gomez-Moreno, and F. Lopez-Ferreras, "Road-sign detection and recognition based on support vector machines," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 2, pp. 264–278, June 2007.

[83] R. Malik, J. Khurshid, and S. N. Ahmad, "Road sign detection and recognition using colour segmentation, shape analysis and template matching," *Proceedings of the International Conference on Machine Learning and Cybernetics*, vol. 6, pp. 3556–3560, August 2007, Hong Kong, China.

[84] J. Greenhalgh and M. Mirmehdi, "Real-time detection and recognition of road traffic signs," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1498–1506, December 2012.

[85] W.-J. Won, M. Lee, and J.-W. Son, "Implementation of road traffic signs detection based on saliency map model," *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 542–547, June 2008, Eindhoven, Netherlands.

[86] R. Kastner, T. Michalke, T. Burbach, J. Fritsch, and C. Goerick, "Attention-based traffic sign recognition with an array of weak classifiers," *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 333–339, June 2010, San Diego, CA, USA.

[87] R. Belaroussi, P. Foucher, J.-P. Tarel, B. Soheilian, P. Charbonnier, and N. Paparoditis, "Road sign detection in images: A case study," *Proceedings of the International Conference on Pattern Recognition*, pp. 484–488, August 2010, Istambul, Turkey.

[88] D. C. W. Pao, H. F. Li, and R. Jayakumar, "Shapes recognition using the straight line Hough transform: Theory and generalization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 11, pp. 1076–1089, November 1992.

[89] S. Houben, "A single target voting scheme for traffic sign detection," *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 124–129, June 2011, Baden-Baden, Germany.

[90] H. Fleyeh and P. Zhao, "A contour-based separation of vertically attached traffic signs," *Proceedings of the Annual Conference of the IEEE Industrial Electronics Society*, pp. 1811–1816, November 2008, Orlando, FL, USA.

[91] L.-W. Tsai, J.-W. Hsieh, C.-H. Chuang, Y.-J. Tseng, K.-C. Fan, and C.-C. Lee, "Road sign detection using eigen colour," *IET Computer Vision*, vol. 2, no. 3, pp. 164–177, September 2008.

[92] G. Loy and A. Zelinsky, "Fast radial symmetry for detecting points of interest," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 959–973, August 2003.

[93] N. Barnes, A. Zelinsky, and L. Fletcher, "Real-time speed sign detection using the radial symmetry detector," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 2, pp. 322–332, June 2008.

[94] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, May 2004.

[95] C. G. Keller, C. Sprunk, C. Bahlmann, J. Giebel, and G. Baratoff, "Real-time recognition of U.S. speed signs," *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 518–523, June 2008, Eindhoven, Netherlands.

[96] X. Baro, S. Escalera, J. Vitria, O. Pujol, and P. Radeva, "Traffic sign recognition using evolutionary Adaboost detection and Forest-ECOC classification," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 1, pp. 113–126, March 2009.

[97] A. R. Rostampour and P. R. Madhvapathy, "Shape recognition using simple measures of projections," *Proceedings of the Annual International Phoenix Conference on Computers and Communications*, pp. 474–479, March 1988, Scottsdale, AZ, USA.

[98] P. Gil-Jimenez, S. Lafuente-Arroyo, H. Gomez-Moreno, F. Lopez-Ferreras, and S. Maldonado-Bascon, "Traffic sign shape classification evaluation. Part II. FFT applied to the signature of blobs," *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 607–612, June 2005, Las Vegas, NV, USA.

[99] A. W. Haddad, S. Huang, M. Boutin, and E. J. Delp, "Detection of symmetric shapes on a mobile device with applications to automatic sign interpretation," *Proceedings of the IS&T/SPIE Electronic Imaging - Multimedia on Mobile Devices*, no. 8304, pp. 1–8, January 2012, San Francisco, CA, USA.

[100] T. Pavlidis, "A review of algorithms for shape analysis," *Computer Graphics and Image Processing*, vol. 7, no. 2, pp. 243–258, April 1978.

[101] ——, "Algorithms for shape analysis of contours and waveforms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-2, no. 4, pp. 301–312, July 1980.

[102] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509–522, April 2002.

[103] O. R. Mitchell and T. A. Grogan, "Global and partial shape discrimination for computer vision," *Optical Engineering*, vol. 23, no. 5, pp. 484–491, October 1984.

[104] C. T. Zahn and R. Z. Roskies, "Fourier descriptors for plane closed curves," *IEEE Transactions on Computers*, vol. C-21, no. 3, pp. 269–281, March 1972.

[105] E. Persoon and K.-S. Fu, "Shape discrimination using Fourier descriptors," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-7, no. 3, pp. 170–179, March 1977.

[106] D. Zhang and G. Lu, "Evaluation of MPEG-7 shape descriptors against other shape descriptors," *Multimedia Systems*, vol. 9, no. 1, pp. 15–30, July 2003.

[107] R. Chellappa and R. Bagdazian, "Fourier coding of image boundaries," *IEEE Transactions onPattern Analysis and Machine Intelligence*, vol. PAMI-6, no. 1, pp. 102–105, January 1984.

[108] C. Singh and P. Sharma, "Performance analysis of various local and global shape descriptors for image retrieval," *Multimedia Systems*, vol. 19, no. 4, pp. 339–357, July 2013.

[109] I. Kunttu, L. Lepisto, J. Rauhamaa, and A. Visa, "Multiscale Fourier descriptor for shape-based image retrieval," *Proceedings of the IEEE International Conference on Pattern Recognition*, vol. 2, pp. 765–768, August 2004, Cambridge, UK.

[110] M. N. Tahir, A. Hussain, and M. M. Mustafa, "Fourier descriptor for pedestrian shape recognition using support vector machine," *Proceedings of the IEEE International Symposium on Signal Processing and Information Technology*, pp. 636–641, December 2007, Cairo, Egypt.

[111] J. Ma, Z. Zhang, H. Tang, and Q. Zhao, "Fast Fourier descriptor method of the shape feature in low resolution images," *Proceedings of the IEEE International Conference on Wireless Communications Networking and Mobile Computing*, pp. 1–4, September 2010, Chengdu, China.

[112] J. Meyer, P. Schnitzspan, S. Kohlbrecher, K. Petersen, M. Andriluka, O. Schwahn, U. Klingauf, S. Roth, B. Schiele, and O. Stryk, "A semantic world model for urban search and rescue based on heterogeneous sensors," *Proceedings of the Annual RoboCup International Symposium*, vol. 6556, pp. 180–193, June 2010, Singapore, Singapore.

[113] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-up robust features (SURF)," *Journal of Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, June 2008.

[114] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 886–893, June 2005, San Diego, CA, USA.

[115] S. Suzuki and K. Abe, "Topological structural analysis of digitized binary images by border following," *Computer Vision, Graphics, and Image Processing*, vol. 30, no. 1, pp. 32–46, April 1985.

[116] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, January 1972.

[117] J. Sklansky, "Finding the convex hull of a simple polygon," *Pattern Recognition Letters*, vol. 1, no. 2, pp. 79–83, December 1982.

[118] S. Pereira and T. Pun, "Robust template matching for affine resistant image watermarks," *IEEE Transactions on Image Processing*, vol. 9, no. 6, pp. 1123–1129, Jun. 2000.

[119] F. Essannouni and D. Aboutajdine, "Fast frequency template matching using higher order statistics," *IEEE Transactions on Image Processing*, vol. 19, no. 3, pp. 826–830, March 2010.

[120] I. Bartolini, P. Ciaccia, and M. Patella, "WARP: Accurate retrieval of shapes using phase of Fourier descriptors and time warping distance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 1, pp. 142–147, January 2005.

[121] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-9, no. 1, pp. 62–66, January 1979.

[122] L.-S. Jin, L. Tian, R.-B. Wang, L. Guo, and J.-W. Chu, "An improved Otsu image segmentation algorithm for path mark detection under variable illumination," *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 840–844, June 2005, Las Vegas, NV, USA.

[123] P. Soille, *Morphological Image Analysis: Principles and Applications*. New York/Heidelberg: Springer-Verlag, 2003.

[124] H. Park and R. Chin, "Decomposition of arbitrarily shaped morphological structuring elements," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 1, pp. 2–15, January 1995.

[125] R. C. Gonzalez, *Digital Image Processing*. New Jersey, USA: Prentice Hall, 2000.

[126] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: A technical overview," *IEEE Signal Processing Magazine*, vol. 20, no. 3, pp. 21–36, May 2003.

[127] W.-C. Siu and K.-W. Hung, "Review of image interpolation and super-resolution," *Proceedings of the Asia-Pacific Signal Information Processing Association Annual Summit and Conference*, no. 6411957, pp. 1–10, December 2012, hollywood, CA, USA.

[128] Y. J. Liang, J. G. Apostolopoulos, and B. Girod, "Analysis of packet loss for compressed video: Effect of burst losses and correlation between error frames," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 7, pp. 861–874, July 2008.

[129] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: A review," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 974–997, May 1998.

[130] S. Kumar, L. Xu, M. K. Mandal, and S. Panchanathan, "Error resiliency schemes in H.264/AVC standard," *Journal of Visual Communication and Image Representation*, vol. 17, no. 2, pp. 425–450, April 2006.

[131] S. Wenger, "H.264/AVC over IP," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 645–656, July 2003.

[132] P. Lambert, W. D. Neve, Y. Dhondt, and R. V. de Walle, "Flexible macroblock ordering in H.264/AVC," *Journal of Visual Communication and Image Representation*, vol. 17, no. 2, pp. 358–375, April 2006.

[133] Y. Zhao, S. C. Ahalt, and J. Dong, "Optimal interleaving for 3-D zerotree wavelet video packets over burst lossy channels," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 333–336, March 2005.

[134] J. Rombaut, A. Pižurica, and W. Philips, "Optimization of packetization masks for image coding based on an objective cost function for desired packet spreading," *IEEE Transactions on Image Processing*, vol. 17, no. 10, pp. 1849–1863, October 2008.

[135] T. H. Vu and S. Aramvith, "An error resilience technique based on FMO and error propagation for H.264 video coding in error-prone channels," *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 205–208, June 2009.

[136] K. Tan and A. Pearmain, "A new error resilience scheme based on FMO and error concealment in H.264/AVC," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1057–1060, May 2011.

[137] H. Hadizadeh and I. V. Bajić, "Burst-loss-resilient packetization of video," *IEEE Transactions on Image Processing*, vol. 20, no. 11, pp. 3195–3206, November 2011.

[138] C. Ying, J. Boyce, and X. Kai, "Frame loss error concealment for SVC," *JVT of ISO/IEC MPEG & ITU-T VCEG, Doc. JVT-Q046*, October 2005.

[139] Q. Peng, T. Yang, and C. Zhu, "Block-based temporal error concealment for video packet using motion vector extrapolation," *Proceedings of the IEEE International Conference on Communications, Circuits and Systems and West Sino Expositions*, vol. 1, pp. 10–14, June 2002.

[140] Y. Chen, K. Yu, J. Li, and S. Li, "An error concealment algorithm for entire frame loss in video transmission," *Proceedings of the IEEE Picture Coding Symposium*, pp. 1–4, December 2004.

[141] B. Yan and H. Gharavi, "Efficient error concealment for the whole-frame loss based on H.264/AVC," *Proceedings of the IEEE International Conference on Image Processing*, pp. 3064–3067, October 2008.

[142] Z. Wu and J. M. Boyce, "An error concealment scheme for entire frame losses based on H.264/AVC," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 4463–4466, May 2006.

[143] Y. Wang, J. Y. Tham, K. H. Goh, W. S. Lee, and W. Yang, "A distance-based slice interleaving scheme for robust video transmission over error-prone networks," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1509–1512, May 2011.

[144] B. E. Bayer, "An optimum method for two-level rendition of continuous-tone pictures," *Proceedings of the IEEE International Conference on Communications*, vol. 1, pp. 2611–2615, June 1973.

[145] Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, *SVC reference software JSVM 9.8*, available: http://ip.hhi.de/imagecom_G1/savce/downloads/SVC-Reference-Software.htm.

[146] V. Varsa, M. M. Hannuksela, A. Hourunranta, and Y.-K. Wang, "Non-normative error concealment algorithms," *ITU-T VCEG, Doc. VCEG-N62*, September 2001.

[147] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell System Technical Journal*, vol. 39, pp. 1253–1266, September 1960.

[148] L. N. Kanal and A. R. K. Sastry, "Models for channels with memory and their applications to error control," *Proceedings of the IEEE*, vol. 66, no. 7, pp. 724–744, July 1978.

APPENDIX

A. MERGE IMAGE ACQUISITION PROTOCOL

This Appendix describes the protocol used for acquiring test images for the MERGE project. The images are used for testing various functions of the MERGE image analysis system.

- Persons involved
  - 1 MERGE staff member
- Equipment/Materials needed
  - Pens or pencils
  - 1 Mobile Telephone with Android OS
    - Built-in camera (1MPx and above)
    - 3G/4G/WiFi data connection
    - GPS
  - 1 Digital Camera with Android OS
    - 3G/4G/WiFi data connection
    - GPS
  - Image Recording Forms
  - External Hard Drive

1) Preliminaries (Internet connection required)

   a) Check Date and Time settings on the Android mobile telephone and the digital camera, and ensure date, time, and time zone are set to automatic (network-provided).

   b) Make sure the Android mobile telephone and the digital camera's batteries are fully charged.

c) Make sure the GPS is enabled on the Android mobile telephone and the digital camera.

d) Verify all equipments/materials above are available.

e) Turn flash feature off on the Android mobile telephone and the digital camera.

f) Note: The Image Taker will need to fill out an Image Recording Form for each hazmat sign.

2) Set up environment

a) Stand in front of the hazmat sign, far enough so that the camera can capture all the content, up to 200 feet from the sign for the Android mobile phone, and up to 500 feet from the sign for the digital camera. Stand preferably perpendicular to the surface containing the sign. Limited angles are permitted (45 degrees), as shown in Figure A.1.

b) Make sure weather conditions do not obstruct the view of the hazmat sign.

c) Make sure there are no objects between the camera and the hazmat sign that partially or completely obstruct the view of the hazmat sign.

3) Taking Images of Hazmat Signs

a) Launch the MERGE application on the Android mobile telephone and the digital camera, and login using the Image Taker's ID and password. If this is the first time that the Image Taker is logging into the application, an Internet connection will be required to connect with the MERGE database on the server. From then on, the Image Taker's credential will be stored on the Android device for future use without an Internet connection.

b) Select the "Capture Image" option from the MERGE main screen. The camera activity is then initialized. Note that a new directory with the name MERGE will be created on the Android device's image gallery, where all the images taken using the MERGE application will be stored. Please refer to

this directory when copying the images to the external hard drive (Section 5a).

c) Prepare for taking the image (position the camera as desired, within the recommended distance and angle from the hazmat sign). Make sure all the contents of the hazmat sign can be seen on the device screen.

d) Take an image of the hazmat sign, trying to hold the device as much as stable. The image can be retaken as many times as needed by tapping on the retake option on the camera activity.

e) Tap on the OK button on the camera activity to save the current image. The image will be automatically uploaded to the server and analyzed. The Image Taker should see a notification dialog with the text "Uploading image..." followed by another notification dialog with the text "Analyzing image...". If no Internet connection is available at the time, a warning dialog with the text "No Internet connection available" will be shown to the Image Taker. However, the image is stored in the Android device, and it can be uploaded and analyzed in the future using the "Browse Image" option from the MERGE main screen. If the image has not been uploaded to the server, check the box "Not Successfully Uploaded" on the Image Recording Form.

f) If no Internet connection is available at the time, a warning dialog with the text "No Internet connection available" will be shown to the Image Taker. In this case, the captured image is stored in the device, and it can be uploaded and analyzed in the future using the "Browse Image" option from the MERGE main screen.

g) Please take different images for the same sign, at different distances (10-150 ft) and angles of view (0-45°), and then write down an Image ID shown on the top bar / pop-up window on the result screen, an approximate Angle of View between your viewpoint and the perpendicular plane of the hazmat sign's surface, and an approximate Distance from your viewpoint to the hazmat sign on the Image Recording Form (e.g., 123456, 15°, and 125 ft).

h) Please take at least one image with No Zoom when using the digital camera, and then check the box "No Zoom" on the Image Recording Form. Also take some images using the Optical Zoom when using the digital camera (NO Digital Zoom), and then check the box "Zoom" and mark on an approximate Zoom Value in a box on the Image Recording Form (e.g., 3/4 of the entire optical zoom range).

4) Completing the Image Recording Form (Figure A.2)

a) Record Date (MM/DD/YYYY), Starting Time (HH:MM:SS), the Make and Model of the device used to capture the images (e.g., HTC Desire) and the Image Taker's Name and Affiliation on the Image Recording Form.

b) Complete the "Ground Truth Information" section on the Image Recording Form with ground-truth information associated with each hazmat sign in the captured image. This includes:

- The Total number of existing hazmat signs in the captured image
- For each existing hazmat sign
  - Hazmat sign number of an existing hazmat sign in the captured image
  - Color(s): color(s) found in the hazmat sign (NOT including hazmat sign frame)
  - UN Identification number (UNID) (Figure A.3(a))
  - Symbol (Figure A.3(b))
  - Class (Figure A.3(c))
  - Text (Figure A.3(d))
  - Comments: Additional information of the hazmat sign that does not fit in the previous fields.

c) Complete the "Image Analysis Results" section on the Image Recording Form with information retrieved from the server after a captured or browsed image has been analyzed. This includes:

- The Image ID of the captured image

- The Total number of highlighted hazmat signs from image analysis
- For each returned hazmat sign
  - Hazmat sign number of a highlighted hazmat sign shown in the result screen
  - Color(s): color(s) shown in the result screen
  - Text: text shown in the result screen
  - No hazmat signs found: Check this box if a dialog containing "No hazmat signs found" is shown to the Image Taker after uploading an image to the server, meaning that no hazmat signs have been found in the current image.

There are two cases of image analysis results, hazmat sign found (left) and not found (right), shown in Figure A.4. Figures A.5 and A.6 show two examples of completed Image Recording Forms for the two different cases.

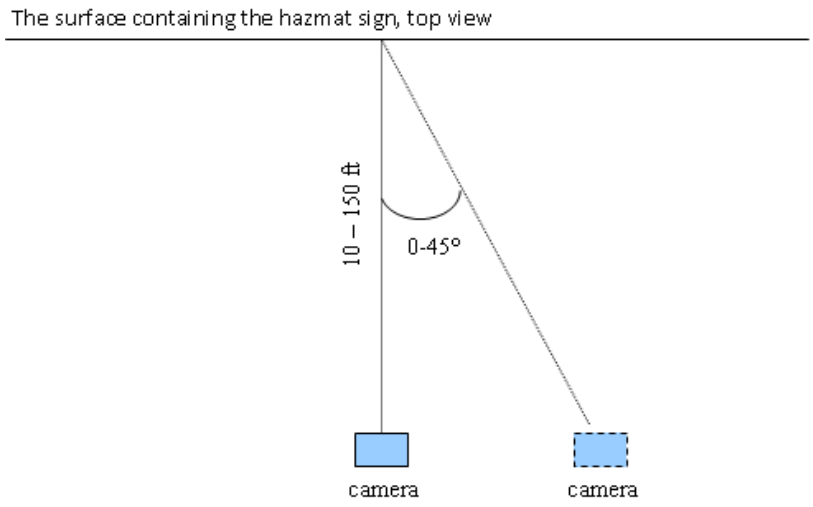


Fig. A.1. Top view of the setup environment.

Image Recording Form

Image Taker Name: **ID:** **Affiliation:**

Date: / / **Starting Time:** : :

Device Make: **Device Model:**

Ground Truth Information			Angle of View		°	Distance		ft
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments
Image Analysis Results			No Zoom [ ]		Zoom [ ]		1/4 1/2 3/4 Full	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found
								[ ]

Ground Truth Information			Angle of View		°	Distance		ft
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments
Image Analysis Results			No Zoom [ ]		Zoom [ ]		1/4 1/2 3/4 Full	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found
								[ ]

Ground Truth Information			Angle of View		°	Distance		ft
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments
Image Analysis Results			No Zoom [ ]		Zoom [ ]		1/4 1/2 3/4 Full	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found
								[ ]

Ground Truth Information			Angle of View		°	Distance		ft
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments
Image Analysis Results			No Zoom [ ]		Zoom [ ]		1/4 1/2 3/4 Full	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found
								[ ]

Fig. A.2. Image recording form for the MERGE project.

(a) UNID (b) Symbol (c) Class (d) Text

Fig. A.3. Hazmat sign identifiers.

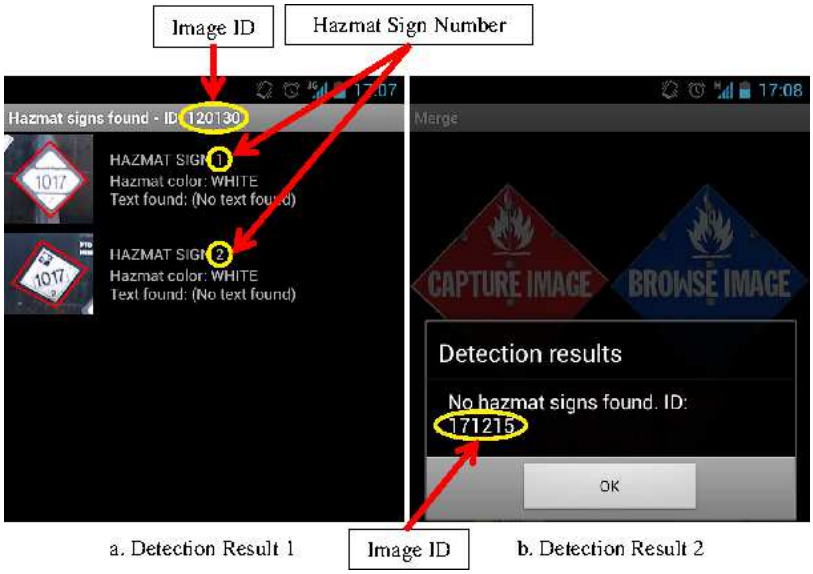


Fig. A.4. Examples and screenshots of the two cases of image analysis results, hazmat sign found (left) and not found (right).

Ground Truth Information			Angle of View		5 °	Distance		80 ft	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments	
120130	1	2	WHITE	1017	Skull	2	No Text		
Image Analysis Results			No Zoom [X]		Zoom [ ]		1/4 1/2	3/4	Full
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found	
120130	1	2	WHITE	N/A	N/A	N/A	No Text	[ ]	

Ground Truth Information			Angle of View		5 °	Distance		80 ft	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments	
120130	2	2	WHITE	1017	Skull	2	No Text		
Image Analysis Results			No Zoom [X]		Zoom [ ]		1/4 1/2	3/4	Full
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found	
120130	2	2	WHITE	N/A	N/A	N/A	No Text	[ ]	

Fig. A.5. Examples of completed image recording form for hazmat sign found in Figure A.4 (left).

Ground Truth Information			Angle of View		15 °	Distance		125 ft	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments	
171215	1	1	RED	1267	Flame	3	No Text		
Image Analysis Results			No Zoom [ ]		Zoom [X]		1/4 1/2	3/4	Full
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found	
171215	1	1						[X]	

Fig. A.6. Examples of completed image recording form for hazmat sign not found in Figure A.4 (right).

VITA

VITA

Bin Zhao obtained the B.S. degree in Telecommunication Engineering and the M.S. degree in Information and Telecommunication Engineering (both with Highest Distinction) from Xidian University, Xi'an, China. He was a Graduate Fellow and Research Assistant of the National Key Laboratory of Integrated Service Networks (ISN), Xi'an, China. He is pursuing the Ph.D. degree in Electrical and Computer Engineering in Purdue University, West Lafayette, Indiana, USA. He was working as a Graduate Teaching Assistant in the School of Electrical and Computer Engineering in Purdue University from 2009 to 2010. He is working as a Graduate Research Assistant in the Video and Image Processing Laboratory (VIPER) under the supervision of Professor Edward J. Delp since 2010. He is a student member of the IEEE, the IEEE Communications Society, and the IEEE Signal Processing Society. His research interests include image analysis, image and video processing, computer vision, object recognition, and machine learning.

Graduate School ETD Form 9
(Revised 12/07)

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Albert Parra Pozo

Entitled
Integrated Mobile Systems Using Image Analysis with Applications in Public Safety

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

EDWARD J. DELP
Chair

JAN P. ALLEBACH

MARY L. COMER

MIREILLE BOUTIN

To the best of my knowledge and as understood by the student in the *Research Integrity and Copyright Disclaimer (Graduate School Form 20)*, this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

Approved by Major Professor(s): EDWARD J. DELP

Approved by: M. R. Melloch 07-02-2014
Head of the Graduate Program Date

INTEGRATED MOBILE SYSTEMS USING IMAGE ANALYSIS WITH APPLICATIONS IN PUBLIC SAFETY

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Albert Parra Pozo

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2014

Purdue University

West Lafayette, Indiana

ACKNOWLEDGMENTS

I would like to thank my first major advisor, Professor Edward J. Delp, for all the challenges he has given me so far, and for believing that I can overcome them. I really value his advice and criticism; it helps me make the most of my academic career.

I am also very thankful for the support and guidance of Professor Mireille Boutin. I appreciate her finding the time to help me with the research, and helping me organize my ideas and see things from different points of view.

I would like to thank the remaining members of my Graduate Committee, Professor Jan P. Allebach and Professor Mary L. Comer.

I want to give special thanks to Dr. Marc Bosch for his advice and support during the time we share at Purdue, and to Andrew W. Haddad for his patience and help in my both my academic and personal life. Special thanks to Dr. Ye He for believing in me and helping me become a better person.

It has been a pleasure being part of the Video and Image Processing Laboratory (VIPER), both for the quality of the research carried out in the lab and for the people involved. Thanks to my current and former colleagues Jeehyun Choe, Neeraj Gadgil, Joonsoo Kim, Deen King-Smith, Dr. Nitin Khanna, Soonam Lee, He Li, Dr. Kevin Lorenz, Dr. Aravind Mikkilineni, Dr. Ka Ki Ng, Thitiporn Pramoun, Dr. Satyam Srivastava, Khalid Tahboub, Kharittha Thongkor, Yu Wang, Dr. Chang Xu, Dr. Meilin Yang, Bin Zhao, and Dr. Fengqing Maggie Zhu.

I would like to thank my parents for supporting my career decisions and always believing in me. Thanks to them for giving me the opportunity to acquire and share knowledge with others.

The gang graffiti images shown in this thesis were obtained in cooperation with the Indianapolis Metropolitan Police Department.

The hazmat sign images shown in this thesis were obtained in cooperation with the Transportation Security Administration.

We gratefully acknowledge their cooperation in GARI and MERGE.

This work was supported by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI000.

TABLE OF CONTENTS

Page

Page

Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

ABSTRACT

Parra Pozo, Albert Ph.D., Purdue University, August 2014. Integrated mobile systems using image analysis with applications in public safety. Major Professor: Edward J. Delp.

One of the roles of emergency first responders (e.g. police and fire departments) is to prevent and protect against events that can jeopardize the safety and well being of a community. Examples include criminal gang activity and the handling and transportation of dangerous materials. In each of these cases first responders need tools for finding, documenting, and taking the necessary actions to mitigate the problem or issue.

The goal of this thesis is to develop integrated mobile-based systems capable of using location-based-services, combined with image analysis, to provide accurate and useful information to the first responders in real time. Two systems have been developed.

The first is a system to track and analyze gang activity through the acquisition, indexing and recognition of gang graffiti images. This approach uses image analysis methods for color correction, color recognition, image segmentation, and image retrieval and classification. A database of gang graffiti images is described that includes not only the images but also metadata related to the images, such as date and time, geoposition, gang, gang member, colors, and symbols. The user can then query the data in a useful manner.

The second is a system that can recognize and interpret hazardous material (hazmat) signs typically displayed by vehicles transporting dangerous materials. This approach uses image analysis methods for hazmat sign interpretation, including shape location detection and color recognition. The detection results are used to query an

electronic version of the Emergency Response Guidebook (ERG) and return information and advice to help first responders. A database of hazmat sign and scene images for forensic analysis is described that includes images and metadata.

1. INTRODUCTION

1.1 Problem Formulation and Challenges

One of the roles of public safety is to prevent and protect against events that can jeopardize the safety and well being of the community. These include criminal gang activity and handling and transportation of dangerous materials. In each of these cases first responders have the potential for finding and documenting evidence in real time. However, the number of actions that can be taken while on the streets are limited. If there is an incident and law enforcement officers need to compare information, they have to communicate with the corresponding police department.

For example, if gang graffiti is spotted by a first responder in an area, the information that can be obtained in situ is very limited. In the best case scenario, the user has expertise with gang graffiti interpretation and carries a camera. The only actions the user can take are reduced to taking an image and writing down some basic context information.

In a different scenario, a truck hauling a hazardous substance must carry a placard that helps identify the material and determine what specialty equipment, procedures and precautions should be taken in the event of an emergency. This information is contained in the Emergency Response Guidebook (ERG), published by the US Department of Transportation (DOT) [1]. As one might expect, the guidebook is large and requires precious time to search an index to determine the best way to handle a particular hazardous material.

The goal of this thesis is to develop integrated mobile-based systems capable of using location-based-services, combined with image analysis, to provide accurate and useful information to the first responders in real time.

1.2 Contributions of This Thesis

In this thesis two integrated mobile systems are described. First, a system for gang graffiti image acquisition and recognition. We called this system Gang Graffiti Automatic Recognition and Interpretation or GARI. GARI includes motion blur prevention and detection, color correction based on light sensor, color recognition based on touchscreen tracing, color image segmentation based on Gaussian thresholding, and content-based gang graffiti image retrieval. We have also investigated the design and deployment of an integrated image-based database system. Second, a system for hazmat sign detection and recognition. We called this system Mobile Emergency Response Guidebook or MERGE. MERGE includes segment detection using geometric constraints, convex quadrilateral detection based on saliency map, and sign location detection based on Fourier descriptors.

The main contributions of GARI and MERGE in the area of image analysis are as follows:

- We presented a motion blur prevention and detection method based on mobile device sensors.
- We presented a color correction method based on mobile device light sensor.
- We described a color recognition method based on touchscreen tracing.
- We presented a color image segmentation method based on Gaussian thresholding, block-wise Gaussian segmentation enhancement, background stripe removal, and connected component reconnection.
- We presented a feature extraction method based on local shape context descriptors from SIFT keypoint locations.

- We presented a gang graffiti content based image retrieval method based on bag-of-words model.
- We presented a segment detection method based on geometric constraints.
- We presented a convex quadrilateral detection method based on saliency map.
- We presented a sign location detection based on Fourier descriptors.

The main contributions of GARI and MERGE in the design and deployment of the integrated image-based database system are as follows:

- We developed an integrated image-based database system where data from users and images is connected to gang graffiti information for analysis and tracking.
- We developed an integrated image-based database system where data from users and images is connected to hazmat sign information for image analysis and forensics.
- We created a web-based interface for first responders and researchers to upload images and browse gang related information by location, date and time, using interactive maps for better visualization. It is accessible from any device capable of connecting to the Internet, including iPhone and Blackberry.
- We created a web-based interface for first responders and researchers to upload images and browse hazardous material information by location, date and time for forensic analysis. It is accessible from any device capable of connecting to the Internet, including iPhone and Blackberry.
- We created Android and iOS applications for first responders on the field to upload images to the server, use image analysis and conduct forensic tasks, browse related information, and use location-based services to populate interactive maps.

1.3 Publications Resulting From This Work

Conference Papers

1. Bin Zhao, **Albert Parra** and Edward J. Delp, "Mobile-Based Hazmat Sign Detection System," *Proceedings of the IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 735-738, December 2013, Austin, TX.

2. **Albert Parra**, Bin Zhao, Joonsoo Kim and Edward J. Delp, "Recognition, Segmentation and Retrieval of Gang Graffiti Images on a Mobile Device," *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, pp. 178-183, November 2013, Waltham, MA.

3. **Albert Parra**, Bin Zhao, Andrew Haddad, Mireille Boutin and Edward J. Delp, "Hazardous Material Sign Detection and Recognition," *Proceedings of the IEEE International Conference on Image Processing*, pp. 2640-2644, September 2013, Melbourne, Australia.

4. **Albert Parra**, Mireille Boutin and Edward J. Delp, "Location-Aware Gang Graffiti Acquisition and Browsing on a Mobile Device," *Proceedings of the IS&T/SPIE Electronic Imaging on Multimedia on Mobile Devices*, pp. 830402-1-13, January 2012, San Francisco, CA.

2. OVERVIEW OF GANG GRAFFITI AND HAZMAT SIGN DETECTION SYSTEMS

2.1 Overview of Gang Graffiti Systems

There are several methods that have been described to identify gang graffiti using feature matching as well as tracking gang graffiti using large databases. This section overviews the current methods describing their advantages and disadvantages. [1] We also compare some of the methods with GARI.

2.1.1 Graffiti Tracker

Graffiti Tracker is a web-based system that began in 2002 [3]. It was designed to help first responders identify, track, prosecute and seek restitution from graffiti vandals. It is primarily used by law enforcement and public works agencies. The database contains more than 2 million manually analyzed graffiti images from 75 cities in two countries and nine states, mainly from the state of California.

The web-based services include graffiti analysis, interactive map browsing, graffiti storing and organization, and graffiti report. Graffiti Tracker provides clients with GPS-enabled digital cameras to generate reports of graffiti activity. The images can then be uploaded through the web interface to the database, where they are manually analyzed by trained analysts within 24 hours of submission.

The GPS coordinates of each image are used to build an interactive map where the user can view activity from individual vandals or monikers to specific crews or gangs. Gang trends or migration can be identified if the volume of graffiti for the same gang or vandal is large. A part from the interactive map, the user can browse the

[1] This chapter is an updated version of a chapter that appeared in [2].

stored graffiti by moniker, gang, type of incident, graffiti surface, or removal method. The information can be used to generate reports based on gang or moniker activity, such as total square feet of damage, locations of the incidents, or frequency of graffiti vandalism over a specific period of time.

2.1.2 TAGRS

Tracking and Automated Graffiti Reporting System (TAGRS) is a system developed by 594 Graffiti, LLC in Irvine, CA in 2010 [4] in cooperation with the Orange County Sheriff's Department (OCSD)/Transit Police Services and Orange County Transportation Authority.

Government employees can access TAGRS through an Internet portal using a smartphone or PDA to input graffiti information including address, amount of damage, images of the graffiti and the date and time it was discovered. Law enforcement officers input their information through a secure intranet. After the graffiti data is entered it is sent to the investigator or analyst designed to handle graffiti offenses. Email support enables investigators to share information. TAGRS also reports on cost analysis and graffiti trends. Training for TAGRS takes about two hours.

The TAGRS program has helped solve more than 300 graffiti cases in Orange County since 2008.

The TAGRS application is provided at no cost, but any implementing agency is responsible for purchasing the hardware and services responsible to utilize and maintain the system. Once a client's device is registered in the TAGRS database it is a cross-mobile platform compatible with iOS, Blackberry and Android.

2.1.3 GRIP

Graffiti Reduction & Interception Program (GRIP) is a graffiti and crime database developed by GRIP Systems in 1999 [5]. Graffiti experts, law enforcement and city management and infrastructure groups designed it.

GRIP allows a contractor to take an image and fill out a form detailing the image, and then send it to GRIP database for instant reading and analysis. An application for GRIP has been created using both iOS and Android. GRIP allows residents to send in images of graffiti from mobile devices or use their computers to email images and graffiti locations. GRIP offers free unlimited use of its database for six months.

Users can do their own data entry with GRIP's guidance, or can choose to use GRIP systems for entry work and analysis. There are multiple access levels including citizen, clerk, law enforcement agency, reader only or contractor.

2.1.4 GTS

The Graffiti Tracking System (GTS) is a system developed by Blue Archer in Pittsburgh, PA in 2005 [6]. It is a centralized, web-based application that enables multiple users to document instances of graffiti crime, manage investigations, track graffiti removal requests and compile actionable intelligence through the Internet.

GTS is designed for use by any organization that is fighting graffiti crime including law enforcement, prosecutors, public works departments, railways, and local and state officials.

Features of the GTS include tracking an unlimited number of graffiti incidents, uploading an unlimited number of photos per incident; intelligent searching of all GTS records; automatic linking of similar incidents to develop actionable intelligence; tracking of unlimited number of suspects, witnesses and contacts per incident; automated notification of new incidents based on user-defined filters; fully customizable drop-down menus to record incident criteria.

2.1.5 GAT

Graffiti Abatement Tool (GAT) is a system developed by the Public Works, Police, and Information Technology departments in Riverside, CA in 2007. This system is not currently commercially available. GAT was developed to coordinate inter-

departmental efforts and address the problem of connecting instances of graffiti to an individual vandal or tagger. It stores and manages images of graffiti with other tabular data. It is claimed that GAT is useful in tracking, prosecuting and suing taggers.

Public Works crews that remove graffiti take a picture of the tag using a GPS camera and complete a customized digital form on the camera including basic information about the incident. The images and data are uploaded onto a server that automatically adds the data to an online database. Graffiti images can be matched with other instances of graffiti by the same tagger.

GAT allows the total cost of graffiti to be estimated. When the Public Works abatement crew removes the graffiti, the cleanup method and materials used as well as how much time was required are entered. The cost associated with prosecuting and suing a tagger in a civil lawsuit is entered by the city attorney. GAT allows for the construction of a chain of evidence for the prosecution. There are more than 200,000 images and associated information in Riverside's central police database, with the number increasing by up to 500 per week. Nearly 83,000 instances of graffiti have been removed since January 2009.

2.1.6 TAG-IMAGE

Tattoo and Graffiti Image-Matching and Graphic Evaluation (TAG-IMAGE) is a system developed by the Federal Bureau of Investigation (FBI) Biometric Center of Excellence (BCOE) in Clarksburg, WV in 2012. The system, which is not currently commercially available, is a collaboration with the Cryptanalysis and Racketeering Records Unit (CRRU) of the FBI's Laboratory Division.

TAG-IMAGE is an image-comparison system designed to help the CRRU match images within its database to determine the significance of tattoos, graffiti or other cryptic symbols for FBI investigative programs dealing with foreign or domestic terrorism, violent crime or gangs.

TAG-IMAGE uses image-to-image technology to match symbols based on appearances. A user emails an image to the CRRU where an analyst enters it into the system. The system then compares the image against images stored in the CRRU database. When the search is completed a CRRU analyst emails a response to the user, including associated details and contact information. The submitted image becomes available for future comparisons by other agencies.

TAG-IMAGE is currently in pilot phase and will become available to local, state, tribal and federal law enforcement and correctional agencies when the pilot phase ends. The BCOE also plans to conduct a small operational pilot program with the National Gang Intelligence Center to determine the feasibility of image-based matching and to gain user feedback.

2.1.7 Graffiti-ID

Graffiti-ID is an ongoing project (since 2009) at Michigan State University [7, 8]. The project is focused on matching and retrieval of graffiti images. There is similar work from the same team on gang tattoo identification, called Tattoo-ID [9–14].

The goal of Graffiti-ID is to identify gang/moniker names related to a graffiti image, based on visual and content similarities of graffiti images in a database. Figure 2.1 shows a block diagram of the system. There are two modules, one for populating the database (offline) and another for querying and obtaining results from the database (online). The offline module includes two processes. First, automatic feature extraction using the Scale Invariant Feature Transform (SIFT) [15]. Second, manual annotation of graffiti images by letters and numbers. This is done on images taken from an external gallery of images with the information stored in a database. The online modules includes manual annotation of input images to filter the database and SIFT feature extraction to obtain keypoint matching.

The image database used is based on the Tracking Automated and Graffiti Reporting System (TAGRS) from the Orange County Sheriff Department in California.

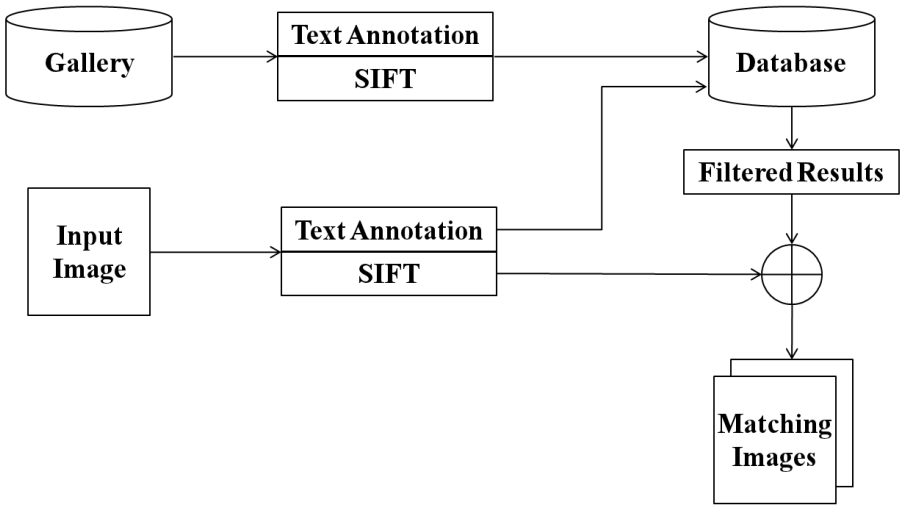


Fig. 2.1.: Block Diagram of the Graffiti-ID System.

The database consists of 64,000 graffiti images the main sources of the images are the Orange County Transportation Authority and crime reports. A subset of 9,367 images were used for evaluation. Each of these images contains up to four information parameters: moniker, gang, date and time, and address.

The Graffiti-ID system was tested using graffiti images from the original database subset. The retrieval accuracy was evaluated using Cumulative Match Scores (CMS) [16]. The graffiti images were used as query for the manual annotation matching step, which returns candidate images from the database that match the text description (presence of letters and numbers). SIFT features from the input image and compared against SIFT features from each of the candidate images. The candidates that best match the SIFT features of the query, given the Cumulative Match Scores, are returned to the user. Table 2.1 shows performance results of the output of the second step. The rank-k accuracy refers to the percentage of queries for which the correctly matched images are found within the k candidate images.

Table 2.1: Accuracy and execution time for various numbers of candidate images from the manual annotation matching step.

Candidate Images	300	500	1,000	9,367
Rank-30 accuracy	63.8 %	65.4 %	66.5 %	64.3 %
Retrieval Time (seconds/query)	12.4 s	20.1 s	39.8 s	415.7 s

2.1.8 Other Work on Graffiti and Tattoos

There exist other methods in the literature that use image analysis techniques on graffiti and tattoo images. In [17] methods for segmenting and retrieving graffiti images are described using global thresholding and template matching. The system consists of two main components: character detection and string recognition and retrieval. The character detection process includes image preprocessing and binarization, text detection and image refinement. The string recognition and retrieval process is further subdivided into two modules: image-wise retrieval and semantic-wise retrieval. The image-wise retrieval includes bounding-box extraction and interest point matching. The semantic-wise retrieval includes bounding-box extraction, character recognition and string matching. The results of the image-wise retrieval and semantic-wise retrieval modules are combined to produce the final output. The experimental results on a database of 194 graffiti images show a retrieval accuracy of 88% when using the proposed bounding box framework.

In [9–14] the authors describe image retrieval approaches for tattoo images, Tattoo-ID. The goal of Tattoo-ID is to create a content based image retrieval system to find images from a database that are related to the query image. The image analysis methods used are very similar to those in Graffiti-ID, including SIFT keypoints and the use of a matching technique to measure visual similarities. The system was tested in a database of 100,000 tattoo images. The retrieval accuracy was 85.6%, with an average retrieval time of 191 seconds on an Intel Core 2, 2.66GHz and 3GB RAM processor.

In [18] the authors propose a tattoo retrieval system using a combination of existing image retrieval techniques. Figure 2.2 illustrates the system. The experimental results on a dataset of more than 300,000 tattoo images show a retrieval accuracy of 85% in the best case. The running times depend on the database used, and range from 145ms to 5 seconds on an Intel i7-930 using 4 cores with 2.8GHz and 8GB of main memory.

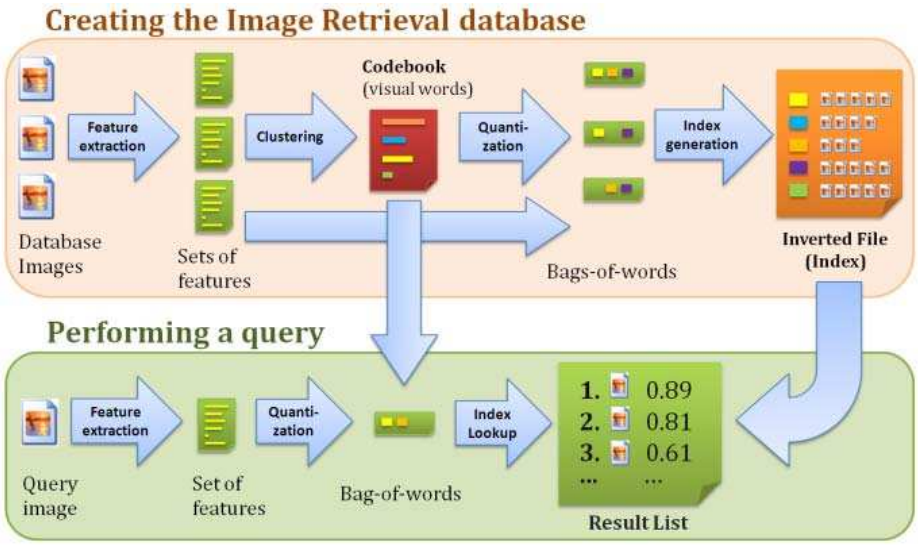


Fig. 2.2.: Block Diagram of The System in [18].

2.1.9 Comparison to GARI

Although our proposed system (GARI) shares some goals with the above systems, our methodology is different. Table 2.2 summarizes a comparison between the features of the various Gang Graffiti Systems described above.

We present a detailed comparison between the most similar systems to GARI: Graffiti-ID and Graffiti Tracker. We then compare the image analysis methods used in other work on graffiti and tattoos to the ones used in GARI.

GARI vs. Graffiti-ID vs. Graffiti Tracker

Both Graffiti-ID and GARI have goals of identifying gangs and gang members based on the graffiti content. Graffiti-ID uses SIFT features between an input image and images from the database. GARI currently uses color recognition techniques, along with metadata information from an image to query the database. GARI uses SIFT features to detect if an image of a same graffiti was already acquired at a specific location. GARI also uses shape techniques to detect graffiti components. By graffiti

Table 2.2: Comparison of features between different gang graffiti systems and GARI.

Feature	**GARI**	**Graffiti-ID**	**Graffiti Tracker**	**TAGRS**	**GRIP**	**GTS**	**GAT**	**TAG-IMAGE**
Used in field	YES	NO	YES	YES	YES	YES	YES	YES
Graffiti location	IN, IL	CA	CA	CA	CA, CO	CA, PA	CA	-
Images in database	1,000	6,000	+4 million	-	-	-	200,000	-
Analysis (time)	Seconds	-	24h	-	-	-	-	-
Analysis (method)	Semiautomatic	Automatic	Manual	Manual	Manual	Manual	Manual	Semiautomatic
Web version	YES	NO	YES	YES	YES	YES	YES	NO
Mobile version	YES	NO	NO	YES	YES	NO	NO	NO
Device	Smartphone	-	GPS Camera	Smartphone	Smartphone	-	GPS Camera	-
Interactive Map	YES	NO	YES	NO	NO	NO	YES	NO
Price	$10,000 server	-	$6,000/year	$15,000 server	$987/year	-	-	-

components we mean the objects and shapes contained in a graffiti image, such as stars, pitchforks, crowns, and arrows.

Both Graffiti Tracker and GARI keep track of gang activity based on GPS tags from the images and the graffiti content. However, all the image analysis in Graffiti Tracker is done manually, while the only user input on GARI is the touchscreen tracing for color recognition.

Graffiti-ID does not exploit the first responders action in the field, such as capture and upload images to a server or browse the database from a mobile device; the analyzed images are on the server. Graffiti Tracker allows users to acquire images only with GPS-enabled cameras they provide and the images have to be transferred to a computer and sent to the server. GARI allows the users to take images with any camera, and the GPS coordinates are automatically extracted from the EXIF data of the image or inserted manually when uploaded to the server (i.e., by GPS coordinates or by address through reverse geocoding [19]). Moreover, GARI has a mobile application that allows the user to take an image with a smartphone and send it to the server in situ. GARI also allows the first responder to browse the database of graffiti. GARI allows the user to upload images to the server through a web-based interface from any device capable of connecting to the Internet.

In Graffiti Tracker image analysis is done manually by trained analysts with the results obtained within 24 hours of submission. GARI currently does the analysis in the field, automatically and in real-time, either on the device or on the server. Graffiti-ID uses, as GARI, SIFT features to match images on the server automatically, but the analysis of the content of the graffiti is done manually, by labeling the image. It just allows labels to be numbers (0-9) or letters (a-z), not symbols or other features such as color.

Graffiti-ID does not provide any type of gang activity tracking, while both Graffiti Tracker and GARI provide interactive maps that allow first responders to browse the database and keep track of specific gangs or individuals. The advantage of GARI is that it also provides additional methods for tracking gang activity, including browsing

the database by radius from specific locations, or by graffiti color. One advantage of Graffiti Tracker is that its database is currently dramatically larger than the GARI database. Therefore, the results retrieved from the Graffiti Tracker database can indicate more accurate gang activity.

In summary, our system combines features from both Graffiti-ID and Graffiti Tracker, and adds more services and functionality. The advantages of our system over Graffiti-ID and Graffiti Tracker are the following. We provide a mobile application to be used by first responders in the field, where they can capture, upload and browse graffiti images from the database. The image acquisition in our system is device independent; virtually any image type from any camera make and model can be uploaded using one of our supported platforms: Android, iOS, and web-based interfaces.

GARI vs. Other Work on Graffiti and Tattoos

The work in [17] is the only method from our review that propose automatic segmentation of the graffiti components from the background. However, while GARI does color image segmentation based on touchscreen tracing, [17] uses local thresholding techniques such as Niblack [20] as a preprocessing step to binarize the image. Also, GARI uses SIFT features for graffiti component image retrieval (see Section 3.7), while [17] uses a template matching method that just considers letters and numbers.

The GARI system uses a vocabulary tree based on a bag-of-words model for content based image retrieval (see Section 3.7). The approaches described in [10, 11] do not use the bag-of-words models and report slower matching and retrieval times than we demonstrate in our experiments (see Section 5).

Finally, although [18] does use a bag-of-words model for image retrieval of gang and gang-like tattoos, the system is not intended for real-time retrieval in mobile-based environments.

2.2 Overview of Hazmat Sign Detection Systems

Although there exist several mobile-based applications that provide easy access to the Emergency Response Guidebook (ERG) guidebook [1, 21], they only provide manually browsing functionality. Several methods in the literature deal with sign location detection and recognition (see Section 4.1), but we are only aware of two other published papers with application to hazmat signs [22, 23].

2.2.1 Hazmat Sign Detection Based on SURF and HBP

In [22] the hazmat sign detection is done using color histogram back-projection (HBP) and Speeded Up Robust Feature (SURF) [24] matching. The method was implemented and tested on an autonomous mobile robot for the 2008 RoboCup World Championship. Histogram back-projection is used to detect regions of interest in the image and remove the background of the scene. A background image without a sign, $h(x, y)$, is used as a ground-truth to isolate the hazmat sign when it appears on the scene and an image of it is captured, $f(x, y)$. This is done by determining the euclidean distance of the color coordinates of each pixel within $h(x, y)$ and the corresponding pixel within $f(x, y)$. A threshold K is used to create a binary mask of the hazmat sign by the use of an indicator function $\delta(x, y) = \{(x, y)\ s.t.\ |f(x, y) - h(x, y)| > K\}$. Several color histograms are then estimated for the U and V channels on the YUV color space, and summed up to create a single histogram $H_o(U, V)$ for every sign on the image. A threshold $\theta(H_o, \epsilon)$ is used for $H_o(U, V)$, resulting in a binary indicator function $\pi_o(U, V)$, which specifies which pixels form part of a sign. The value of ϵ is manually set to 0.05. Finally, morphological filters are used to segment the masked regions from the background and create one or more regions of interest to be used as inputs to the matching process using SURF features.

SURF matching is used to find interest points and retrieve images from a database. After the region of interest is determined from the image containing a hazmat signs, multiple interest points are found using SURF. Interest points surrounding regions

that overlap the region of interest are discarded, since the do not provide enough information about the sign. For the remaining interest points, their corresponding feature vectors are matched against all features of all images in a database corresponding to the colors found on the first step.

The experiments were done using a stereo camera system consisting of two cameras with a resolution of 1024 $\times$ 768 pixels. The tests consisted of detecting five different hazmat signs in 240 images. The images were taken at 1, 1.5 and 2 meters, with a maximum distortion of 30°. The results show a detection accuracy of 92% from 1 meter, 52% from 1.5 meters, and less than 20% from 2 meters. The running time ranges from 1 to 1.6 second on a 2.7GHz Intel CPU.

2.2.2 Hazmat Sign Detection Based on HOG

In [23] hazmat sign detection using sliding windows and Histogram of Oriented Gradients (HOG) [25] is described. The method was implemented and tested on a wheeled USAR robot for the 2010 RoboCup World Championship.

The authors use the sliding window approach to exhaustively scan every pixel over a range of positions and scales, with steps of 8 pixels and relative scale factors of 1.05. For each position and scale a discriminative Support Vector Machine (SVM) classifier is used to make binary decisions about the presence or absence of an object. In order to describe the contents of the image at each particular location a HOG descriptor is used along with color histograms in the Lab color space to distinguish between multiple hazmat signs. For each hazmat sign hypothesis of the HOG based detector, the color histogram is used to do the final classification by applying a k-nearest neighbor approach in combination with χ^2-distance.

The experimental results show a recognition rate of 37.5% using histograms based on entire sliding windows and a recognition rate of 58.3% using sub-region based histograms. Region-based histograms provide better representation of the image since

they are capable of capturing the spatial distribution of colors within the detection window.

2.2.3 Comparison to MERGE

Although all methods above are deployed on mobile environments, MERGE is intended for real-time use by first responders, while [22] and [23] were intended for use in a very specific context. The sign detection method proposed in [22] uses a ground-truth image of the background to aid in detection when the hazmat sign appears. This is not a feasible assumption in MERGE, since the first responders are expected to take images of hazmat signs in a large variety of scenarios. In [23] a dataset of 1,480 daylight images is used for both people and hazmat sign recognition. However, the authors do not specify how many images contain hazmat signs, or at what distances the signs are located. They do not provide information about the resolution of the images or the cameras used for acquisition. In MERGE no assumptions on the background are made in order to detect the sign. Instead, color information is used to detect candidate regions using a saliency map model.

Once the hazmat sign is detected [22] uses image matching based on SURF features, and [23] uses HOG and color histogram descriptors, both being very time consuming task. This step is not done in MERGE. Currently, the color of the hazmat sign is considered to be uniform, and the detection is made at different color channels. The recognition of non-uniformly-colored placards is presented as part of the future work (see Section 6).

The goal of MERGE is to be able to detect hazmat signs at long distances (up to 500 feet). Our experimental results show successful detections in some cases at more than 100 feet. However, the experiments in [22] can only be considered successful at 1.5 meters, and the accuracy reported by [23] is very low. Finally, the execution time of the sign detection method in MERGE is 0.84 seconds on average, faster than the sign detection method in [22]. No execution time is reported in [23].

2.3 Proposed Systems

Two systems have been developed. First, a system to track and analyze gang activity through the acquisition and recognition of gang graffiti images. This approach uses image analysis methods for color recognition, image segmentation, and image retrieval and classification. A database of gang graffiti images is maintained on a server, and includes not only the images, but metadata related to them, such as date and time, geoposition, gang, gang member, colors, or symbols. The user can then query the data in a useful manner. We call this system Gang Graffiti Automatic Recognition and Interpretation or GARI [26] [2].

Second, a system to recognize and interpret hazardous material (hazmat) signs typically displayed by vehicles transporting dangerous materials. This approach uses image analysis methods for hazmat sign interpretation, including shape detection based on saliency maps, color recognition and sign interpretation. The detection results are used to query an electronic version of the ERG and return information and advice to help first responders. We call this system Mobile Emergency Response Guidebook or MERGE [27].

2.3.1 GARI

Gangs are a serious threat to public safety throughout the United States. Gang members are continuously migrating from urban cities to suburban areas. They are responsible for an increasing percentage of crime and violence in many communities. According to the National Gang Threat Assessment, approximately one million gang members belonging to more than 33,000 gangs were criminally active within the United States as of April 2011 [28], an increase of 13,000 since April 2008 [29]. Criminal gangs commit as much as 80% of the crime in many communities according to law enforcement officials. Gang graffiti is their most common way to communicate

[2] Parts of the work on GARI was done with my Purdue colleagues Andrew Haddad and Professor Mireille Boutin.

messages, including challenges, warnings and intimidation to rival gangs. It is an excellent way to track gang affiliation and growth or to obtain membership information.

Our goal is to develop a system, based on a mobile device such as a mobile telephone, capable of using location-based-services, combined with image analysis, to automatically populate a database of graffiti images with information that can be used by law enforcement to identify, track, and mitigate gang activity. The first step towards this goal was to create a system that includes the ability to acquire images in the field using the camera in a mobile telephone and a networked back-end database system that uses the metadata available at the time the image is acquired (geoposition, date and time) along with some basic image analysis functions (e.g. color features) [2].

The next step is to extend the image analysis to include segmentation, matching, retrieval and classification of gang graffiti images and gang graffiti components. By gang graffiti components we mean the objects and shapes contained in a gang graffiti image, such as stars, pitchforks, crowns, and arrows.

Apart from being able to send and retrieve multimedia data to the database, the first responder can take advantage of location-based-services. The information in the database of gang graffiti can be queried to extract information based on parameters such as date and time of capture, upload or modification of the graffiti image, or radius from a given location. The data includes not only the images, but information related to it, such as date and time, geoposition, gang, gang member, colors, or symbols.

We have implemented these features both as applications for Android and iOS [3] hand-held devices and as a web-based interface for any device capable of connecting to the Internet (e.g., desktop/laptop computer, Blackberry).

GARI also includes features for gang tattoo analysis [30]. By providing first responders with this capability, the process of identifying and tracking gang activity can be more efficient.

[3]The iOS application was developed with my Purdue colleague Joonsoo Kim.

System Overview

Figure 2.3 illustrates a block diagram of the GARI system. It shows the various services available, both on the device (no network connection required) and on the server (network connection required). These services include capturing images of gang graffiti, automatic analysis and labeling (such as geoposition, date/time, and other EXIF (Exchangeable Image File Format) [31] data obtained from the image), uploading images to the database of gang graffiti, and querying the database to filter and browse its contents.

Figure 2.4 illustrates the modules of our image analysis system. Note that the modules in bold are currently implemented on the server.

When a first responder uses the mobile device to capture an image we use a customized camera with blur motion prevention (Section 3.2). The image is color corrected on the device using data from the light sensor (Section 3.3) and the user is given several options. The image can be uploaded to the server and added to the database of gang graffiti. If so, we extract EXIF data from the image, such as geoposition and date and time, in order to identify the image and its location. The color recognition module allows the user to detect the color of a graffiti component by tracing a path using their finger on the device's touchscreen (Section 3.4). The color recognition is done entirely on the device and extra data is obtained for color image segmentation from the server (Section 3.5). The content-based image retrieval module finds matches for each segmented graffiti component (Sections 3.7 and 3.6). The captured image can be used to find similar images in the database using the scene recognition module (Section 3.7). The results from the scene recognition and the graffiti component retrieval are sent back to the user. All the data from the different modules can be sent to the server along with the graffiti image, and added to the database to be browsed or analyzed in the future.

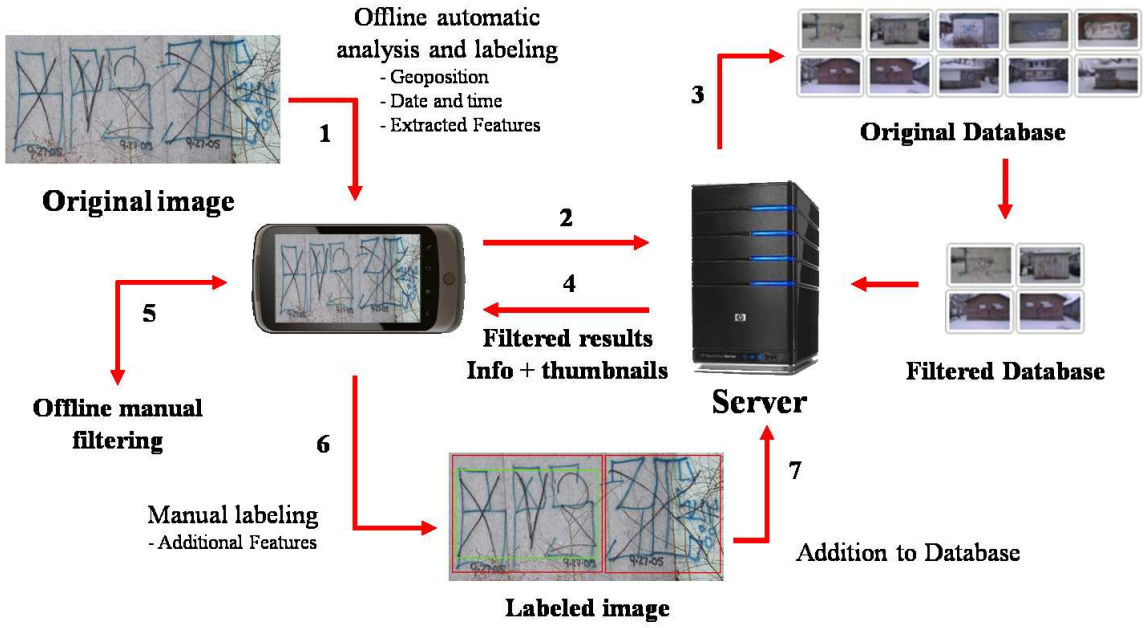


Fig. 2.3.: Block Diagram of the GARI System.

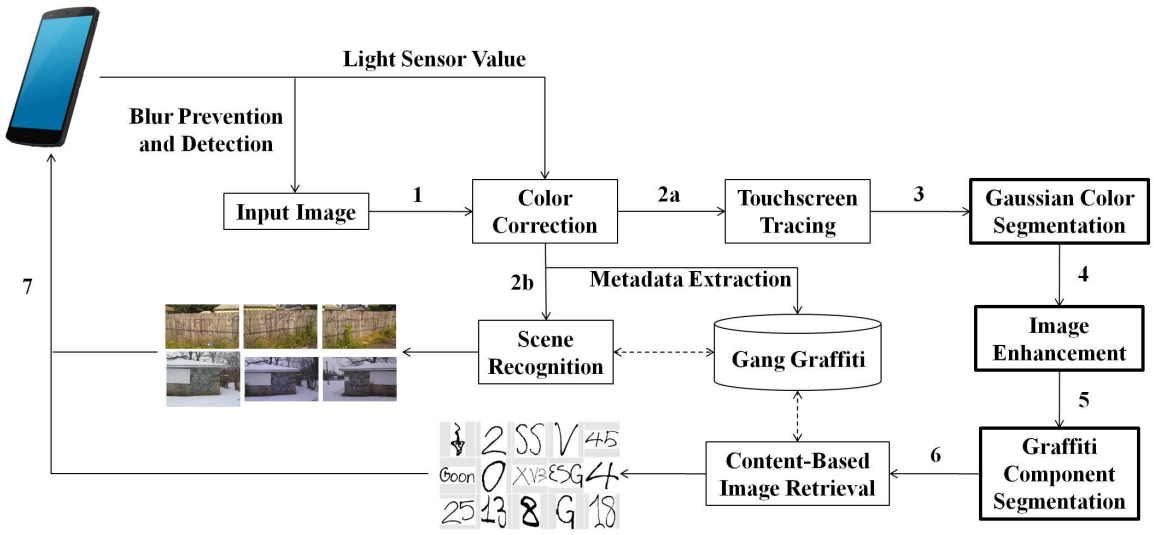


Fig. 2.4.: Modular Components of the GARI System.

Interpretation of Gang Graffiti

Gangs have used street graffiti to communicate with each other for a long time [32, 33]. It is their most common way to communicate messages, including challenges,

warnings or intimidation to rival gangs. If graffiti are correctly interpreted, they are a great source of information that can be used to track gang affiliation and growth, or to obtain membership information.

It is worth noting the differences between "graffiti" terms that we use throughout this thesis.

- **Gang**: We use the word gang to refer to a street gang, defined by [34] as a "self-formed association of peers, united by mutual interests, with identifiable leadership and internal organization, who act collectively or as individuals to achieve specific purposes, including the conduct of illegal activity and control of a particular territory, facility, or enterprise".

- **Gang member**: To be distinguished from a tagger. Gang members paint graffiti to mark territory, threaten other gangs or honor other gang members. In contrast, **taggers** paint graffiti to defy authority, or to obtain recognition or notoriety.

- **Gang graffiti**: To be distinguished from tagging. Gang graffiti are simple and usually monochromatic. In contrast, **tags** are artistic and colorful.

- **Component**: Any of the separable elements in a graffiti, such as symbols, acronyms, or numbers.

- **Blob**: Area of the graffiti containing only one component. Useful to identify relative positions of components to each other in the same graffiti.

- **Clique**: Subset of a larger gang with their own name, which may have connection to the gang's neighborhood (e.g., street name, geographic location). Cliques are local, while gangs extend nationally or internationally. Also known as **factions** or **crews**.

- **Turf**: Slang for territory, or area of influence, specific in this thesis to gangs. Term used when talking about a fight between gangs for territory or power, also

known as a turf war, usually with the objective to gain control over the drug market in a specific area.

In the following subsections we describe how to interpret gang graffiti from its contents, including colors, shapes and structure. We also describe how gangs and gang members can be tracked from the graffiti contents and their location. Finally, we illustrate some examples on how a first responder can do the interpretation and tracking easier and faster. Note we are not claiming in this thesis to be an expert in the interpretation of gang graffiti. Our knowledge is limited. We are relying on law enforcement experts for the GARI project. [4]

Some Examples of Interpretation

Gang graffiti can be considered a low-level language used by gangs to communicate with each other. The alphabet of this "language" consists not only of letters (Aa-Zz) and numbers (0-9) but also of symbols (e.g., stars, crowns, arrows) and colors. The contents of gang graffiti are simple and straightforward. Gangs usually paint handwritten graffiti using a single color (perhaps two at most). Gang graffiti do not contain complete sentences, but words, short phrases, abbreviations and acronyms (e.g., gang and gang member names, street names and numbers). As is the syntax in a regular language, the relative position and alignment of each component is important in the general structure of the graffiti. The syntax in gang graffiti is two-dimensional. For example, the meaning of a symbol is different if it is painted at the top right of a graffiti or if the symbol appears upright or upside down. Figures 2.5 and 2.6 illustrate some examples of gang graffiti alphabet, syntax, and color.

[4]The images shown in this thesis were obtained in cooperation with the Indianapolis Metropolitan Police Department (IMPD). We gratefully acknowledge their cooperation in GARI.

(a) Shape (b) Numbers

(c) Symbols (d) Colors

(e) Letters (f) Position and Alignment

Fig. 2.5.: Examples of Graffiti Elements.

We will use Figure 2.5 as examples for interpreting gang graffiti. Figure 2.5a is a black gang graffiti. This particular color does not eliminate any gang from being the author of the painting. The 6-point star refers to the Folk Nation, one of the two "nations" to which most gangs belong. Each point means: love, life, loyalty, wisdom, knowledge, understanding. The numbers on both the left and the right of the star, 7

(a) Mexicanos Malditos Sureños 13

(b) 18th Street Gang (black) VS Sureños 13 (red)

Fig. 2.6.: Examples of Graffiti Color Recognition.

and 4, refer the 7th and 4th letters of the alphabet, G and D, respectively. That is, the Gangster Disciples gang. The three-pointed pitchfork is another sign of the Folk Nation. In this particular case, two upright three-pointed pitchforks make a total of six points, making reference to the 6-pointed star. Moreover, the inscription below the star makes reference to the clique with the street name, *2-8th st* or 28th street, and the nickname of the gang member who painted the graffiti, *Ruthless*.

Figure 2.5b is a black gang graffiti containing the name of a clique, as usual taking its name is taken from the street where they operate. In this case, it refers to the 42nd Street Gang from Indianapolis. The color itself does not indicate anything concerning which gang this clique may belong to.

Figure 2.5c is a blue gang graffiti with a 6-point star similar to the one in Figure 2.5a. The blue color is used by the Gangster Disciples (and others). The numbers on the sides of the star, along with the additional letters at its bottom make it clear that

this graffiti makes reference to the Gangster Disciples. The number 6 in the center of the star is also an extra remainder of the Folk Nation.

Figure 2.5d is a red and black gang graffiti containing the name of a gang/clique in red, Goon Squad (also spelled *Goon Sqaud* or *Goun Sqoud*). This gang/clique name is very common, since it originally refers to a group of thugs or mercenaries associated with violent acts. With the little information from this graffiti it is not possible to determine which gang they belong to or if they are a gang themselves. However, the use of the red color seems to be related to the People Nation, although there are gangs from the Folk Nation that also use the same color. Below the gang name we find the name of the neighborhood where the gang operates (i.e., *Brightwood 2-5st* or Brightwood 25th Street, Indianapolis) in black. The two down arrows at each side of the gang name express turf dominance. The inscription at the very bottom, also in black, appears to be the nickname of the gang member who painted the graffiti, *7MOB*, also known as "Brightwood 7 M.O.B. Bitch." There is an additional down arrow, again expressing turf dominance of this particular gang member.

Figure 2.5d is a simple black gang graffiti containing the acronym *ESG*, referring to the East Street Gang in Indianapolis.

Figure 2.5f is a multicolor gang graffiti. It seems the blue graffiti was painted over the black graffiti. The black graffiti is very similar to the one in Figure 2.5c, belonging to the Gangster Disciples. The 28th Street clique name, along with the nickname *Ruthless*, are also painted next to the 6-point star. The blue graffiti contains the name of a different clique, the *25th Hillside*, from Hillside Avenue in Indianapolis. The inscription at the very bottom, in blue, could make reference to an insult to the gang or gang member who painted the black graffiti originally, however the upside-down 5-point star indicates disrespect for the People Nation. Therefore, both the black and the blue graffiti have been painted by gang members of Folk Nation's gangs, and the blue inscription to the left of the upside-down 5-point star is the nickname of a gang member of the 25th Hillside clique, from the Folk Nation.

2.3.2 MERGE

Hazardous materials can react differently to environmental stimuli and cause problems in accidents and emergency situations and therefore makes these materials particularly dangerous to civilians and first responders. A federal law in the US requires vehicles transporting hazardous materials be marked with a standard sign (i.e., a "hazmat sign") identifying the type of material the vehicles is carrying [35]. These signs have identifying information described by the sign shape, color, symbols, and numbers.

Our goal is to develop a system, based on a mobile device such as mobile telephone, capable of using location-based-services, combined with image analysis, to automatically detect and interpret hazmat signs from an image taken by a first responder[5].

This system includes the ability to acquire images in the field using the camera in a mobile telephone and a networked system that uses the metadata available at the time the image is acquired (geoposition, date and time) along with image analysis functions to interpret one or multiple hazmat sign on the same image.

The interpretation of the signs includes the association of the sign contents to a guide page on the ERG [1,21]. The information in the book determines what specialty equipment, procedures and precautions should be taken in the event of an emergency related with such chemical component.

Apart from being able to send and retrieve multimedia data to the server, the first responder can take advantage of location-based-services. The location information acquired through the mobile phone can be used along with the interpretation of the hazmat sign to provide the first responder with the best way to handle a particular hazardous material. This is done by projecting an action radius on a multimedia map on the hand-held device, so that the first responder can take the necessary actions to evacuate the affected area. The action radius takes into account real-time weather

[5]Parts of this work was done with my Purdue colleagues Bin Zhao, Andrew Haddad, He Li, Kharittha Thongkor and Professor Mireille Boutin.

information (i.e. wind speed and direction) to provide more accurate evacuation information.

We have implemented these features both as an application for Android hand-held devices and as a web-based interface for any device capable of connecting to the Internet (e.g., desktop/laptop computer, iPhone, Blackberry).

By providing first responders with this capability, the process of identifying and protecting citizens against hazardous materials can be faster and more efficient.

System Overview

Figure 2.7 illustrates a block diagram of the MERGE system. It illustrates the various services available, both on the device (no network connection required) and on the server (network connection required). These services include capturing images of hazmat signs, uploading images to the server for automatic analysis, and querying an internal database containing a digitized version of the ERG [1, 21].

There are two basic operation modes: analysis of a new image and internal database browsing. The first mode includes capturing or browsing for an existing image on the hand-held device, uploading the image to the server and using sign detection and interpretation methods (Section 4). The results sent back to the user include the detected hazmat signs and a link to a guide page from the internal database containing the necessary information in case of an emergency. The second mode includes browsing an internal database to obtain information about the hazmat sign. The internal database can be browsed by UN number, class, symbol, or color (Section 4.5).

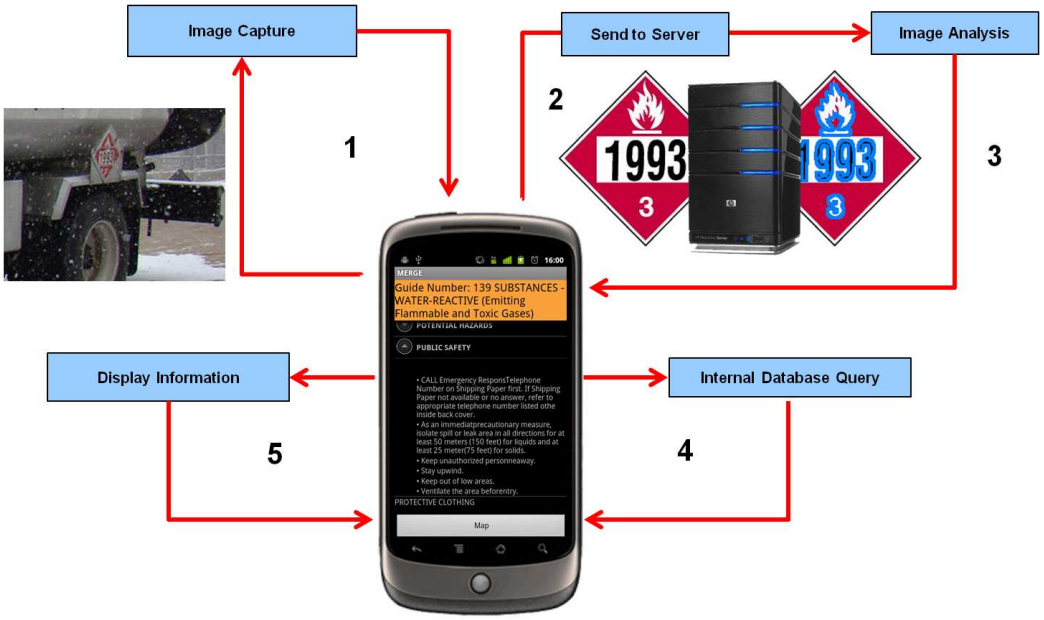


Fig. 2.7.: Block Diagram of the MERGE System.

Interpretation of Hazmat Signs

Hazmat signs are characterized both by their shape and contents. Figure 2.8 illustrates possible shapes for hazmat signs, from which we only consider the diamond-shaped signs, or placards. Inside the placard there are four elements that uniquely identify the chemicals inside the container. Figure 2.9 illustrates three of the elements.

- **UNID**: The United Nations Identification number (UNID) consists of a four-digit number used world-wide in international commerce and transportation to identify hazardous chemicals or classes of hazardous materials. UNID numbers range from 0001 to about 3500 and are assigned by the United Nations Committee of Experts on the Transport of Dangerous Goods. the UNID provides the user a direct link to the ERG guide page containing information on the placard of interest.

- **Symbol**: The graphics and text in the placards representing the dangerous goods safety marks are derived from the UN-based system of identifying dan-

gerous goods. A comparison of symbols in the database will inform the user which guide page is associated with the symbol in the image. The possible symbols (shown in Figure 2.10) are: Corrosive, Explosive, Flammable, Gases, Infectious, Oxidizing, Pollutant, Radioactive, Toxic.

- **Class number**: Following the UN Model, the Department of Transportation divides regulated hazardous materials into nine classes, some of which are further subdivided. The class number on the placard provides the user a number of possible ERG guide pages. The possible classes are: Explosives, Gases, Flammable Liquids, Flammable Solids, Oxidizing Substances, Toxic Substances, Corrosive Substances, Miscellaneous Hazardous Materials.

- **Color**: The color of the hazmat also gives information about the chemical being hauled. The hazmat colors are red, blue, yellow and white. Red is for flammability, blue indicates health hazards, yellow is for reactivity and white is for personal protection. Figure 2.11 shows some possible combinations of colors on hazmat signs.

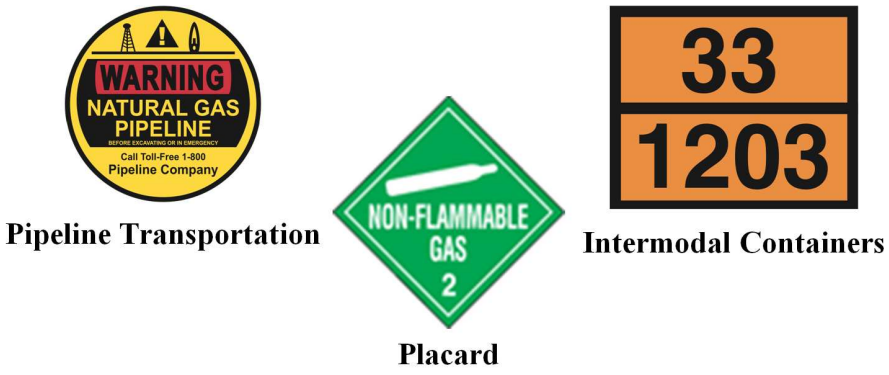


Fig. 2.8.: Possible Shapes of Hazmat Signs.

Fig. 2.9.: Elements That Uniquely Identify a Hazmat Sign. From Left to Right: UNID, Symbol, and Class Number.

Fig. 2.10.: Possible Symbols On a Placard.

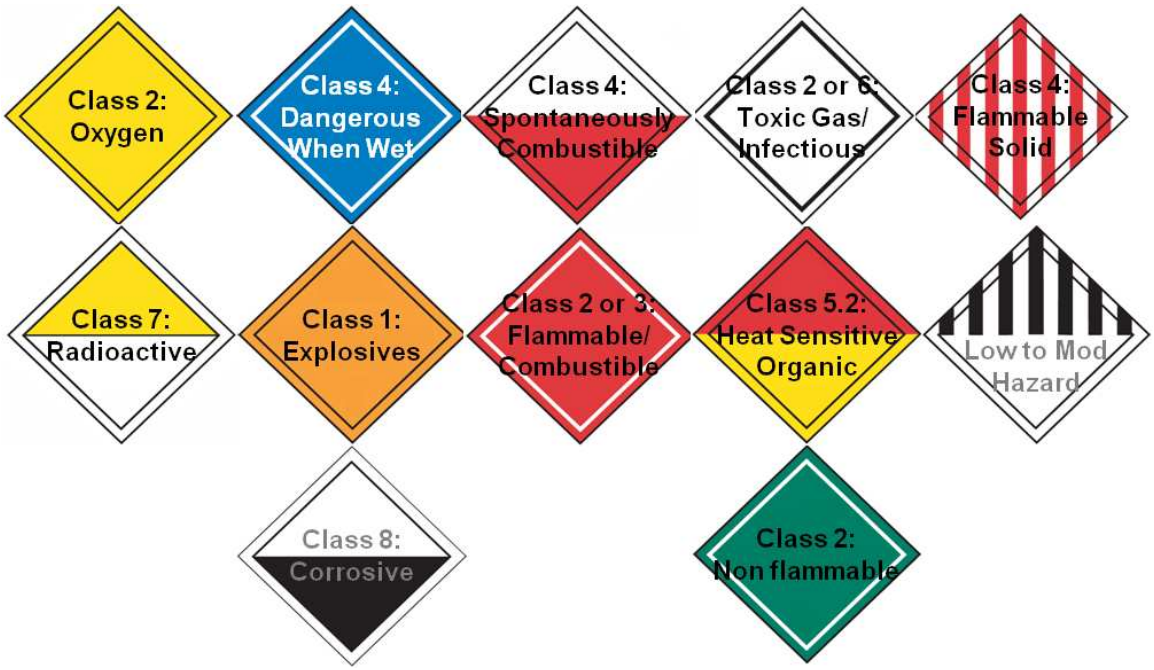


Fig. 2.11.: Possible Colors On a Placard.

3. GANG GRAFFITI AUTOMATIC RECOGNITION AND INTERPRETATION (GARI)

3.1 Review of Existing Methods

In this section we review some relevant literature in the areas of blur detection, color correction, color recognition, color image segmentation, graffiti content analysis, image feature extraction, and image retrieval and classification.

3.1.1 Blur Detection

Image blur is one the most quality degrading distortions in images [36]. It may be caused by out-of-focus, relative motion between the camera and the objects, or inaccurate acquisition settings [37]. In particular, motion blur is one of the main source of blurriness in digital images [38]. Motion blur is caused by camera shake and other movements, and it can cause serious image degradation that can compromise the recognition of objects in the image. Since cheap camera modules in mobile device cameras are not robust to handshaking or low light conditions both hardware and software approaches have been proposed to overcome this problem [39].

Hardware approaches include stabilized lens [40] or Shift-CCD/CMOS used to compensate camera motion [41, 42]. However, this approaches require the use of special equipment, which makes them less suitable for general use. Software approaches can be divided into two categories: frequency domain methods (e.g., computing a transform) and spatial domain methods (e.g., analyzing edges) [43, 44].

In [45] the authors propose a method to measure the global blur using the Discrete Cosine Transform (DCT) [46] information in the image. In order to be as independent

as possible from the image content, their method looks at the distribution of null coefficients instead of the values themselves. This is based on the fact that blurred images tend to have a large number of their high frequency coefficients set to zero. The quality measure is obtained by using a weighting grid that gives more importance to the coefficients on the central diagonal of the DCT matrix, since they better characterize global (circular, non-directional) blur. This method is very sensitive to uniform background and over-illuminated images. Also, its design was aimed at detecting and quantifying only out-of-focus blur.

In [47] the authors propose a blur detection and quantification method based on edge type and sharpness analysis using the Haar-Wavelet Transform (HWT) [48]. The method takes advantage of the ability of the HWT in discriminating edge types, and can detect both out-of-focus and linear-motion blur. Edges are classified into four types: Dirac-Structure, Roof-Structure, Astep-Structure, and Gstep-Structure, the last two being derived from the Step-Structure type. A HWT with three levels of decomposition is first determined, an edge map is then constructed in each scale. After that, this edge map is partitioned, and local maxima in each window are found. If the number of Dirac and Astep structures occurrences are above a threshold, the image is considered blurred.

In [49] a no-reference blur metric based on edge length is proposed. First, a Sobel operator [50] is used to detect edge locations on the luminance component of the image. Then, the edge lengths corresponding to the distance between the starting and ending positions of the edge are computed. The global blur measure is obtained by averaging the lengths over all edges found. This method only considers Gaussian blur. In [44,51] a low complexity blur metric based on Cumulative Probability of Blur Detection (CPBD) which utilizes probability distribution of edge widths is described.

Generally, spatial domain methods are more efficient than frequency domain methods for blur detection, as they do not require an additional transformation to another

domain.

A third category may be added to consider methods that use software approaches based on data obtained from hardware. In [52] inertial sensors (i.e., accelerometers and gyroscopes) built into the mobile device is used to detect motion trajectory of the camera during exposure and then estimate and remove blur from the resulting image. In [53] a "shake metric" technique for detecting camera shake using the mobile device built-in accelerometer to alert blind users in real-time to hold the camera more steadily is described. They do not propose any blur detection method to deal with out-of-focus or low light conditions.

3.1.2 Color Correction

One of the main properties of gang graffiti is its color. If the graffiti contents do not provide any useful information we can use color to filter gang cliques in the area and narrow the possibilities. When color correcting an image we alter its color intensities to match a reference color under a reference illumination [54, 55].

A common approach is to first estimate the scene illumination and then create a mapping between the estimate and the reference [56]. The concept is that both the intrinsic properties of a surface and the color of the illuminant have to be estimated, while only the product of the two (i.e. the actual image) is known. Current approaches can be divided into three categories: static methods, gamut-based methods, and learning-based methods [56, 57].

Static methods use a fixed parameter setting. In [58] using the gray-world assumption [59]: "the average reflectance in a scene under a neutral light source is achromatic" is described. Therefore, any deviation from achromaticity in the average scene color is caused by the effects of the illuminant. The color of the light source is estimated by segmenting the image and computing the average color of all segments.

In [60] a framework known as "gray edge" that uses higher order image statistics such as first and second image derivatives is presented. This method archives the same results as [58] by realizing that the gray-world methods are special instantiations of the L^∞ Minkowski norm. In [61] a fiducial marker with 12 color patches that they place in the image to estimate the illumination parameters is described. In [62] the use of a mobile device touchscreen to obtain the user input by displaying a captured image alongside a color grid of commonly occurring colors is investigated. The user specifies color pairs (i.e. patches in the scene and veridical colors on the grid), which are used to estimate the white point. The estimated white point is then used to construct a diagonal transform to determine the camera output under a desired illuminant.

Gamut-based methods are based on the assumption that in real-world images, for a given illuminant, one observes only a limited number of colors [63]. This limited set of colors that can occur under a specific illuminant is known as the canonical gamut and is determined in a training phase by observing as many surfaces, under one known light source (known as the canonical illuminant), as possible.

In [64, 65] a gamut mapping method that takes as input an image taken under an unknown light source along with the precomputed canonical gamut and estimates the gamut of the unknown light source by assuming that the colors in the input image are representative for the gamut of the unknown light source is presented. In [66, 67] the gamut mapping approach by adding dependence on the diagonal modelmis extended. Under the assumption of the diagonal model, a unique mapping exists that converts the gamut of the unknown light source to the canonical gamut. However, if the diagonal model does not fit the input data accurately, then it is possible that no feasible mapping can be found. This situation is avoided by incrementally augmenting the input gamut until a nonempty feasible set is found.

Learning-based methods estimate the illuminate using a model that is learned on training data.

In [68] a color-by-correlation method that replaces the canonical gamut with a correlation matrix is discussed. One correlation matrix is obtained for every considered illuminant and then used to obtain a probability for every considered light source. Using these probabilities a light source is selected using maximum likelihood [68] or Kullback-Leibler divergence [69]. Other methods use low-level statistics based on the Bayesian formulation [70, 71] and conditional random fields [72]. They model the variability of reflectance and light source as random variables. The illuminant is then estimated from the posterior distribution conditioned on the image intensity data.

Note that all the methods mentioned above use a single image from a regular digital camera to estimate the illuminant. There exist other methods that use additional images [73], specially designed devices [74] or video sequences [75].

3.1.3 Color Recognition

Gang graffiti are often sprayed in non-uniform surfaces, which makes them difficult to distinguish from the background. Since our system is deployed on a mobile telephone, we take advantage of the touchscreen capabilities of modern mobile devices to aid the recognition of color in gang graffiti images.

Since the first capacitive touchscreen was introduced in 1965 [76] multiple applications have been developed for the use of this device. Some examples include interactive surfaces such as sensitive walls [77], cooperative sharing and exchange of media [78], and freehand manipulation [79]. Most modern mobile devices use touchscreens with tactile feedback to interact with the user. This is used to control the device behavior with gestures [80]. The most common application is the virtual keyboard, which is known to be able to improve the performance of text entry with

respect to physical keyboards [81]. The touchscreen can be used to detect a path drawn with the finger on the screen for image analysis such as color recognition. This technique has been previously used to aid the acquisition of morphometric data from pulmonary tissues [82].

Color recognition techniques using tactile feedback use thresholds based on perceptual attributes of specific color spaces [83]. The perceptual thresholds (also known as discrimination thresholds) have been widely studied for human observers [84]. However, some methods do use thresholds based on human perceptibility, but use application-based thresholds. For example, some skin detection methods use an adaptive skin color filter to detect color regions, by setting thresholds in both RGB and HSV color spaces [85, 86].

3.1.4 Color Image Segmentation

In order to interpret the contents of a gang graffiti, we first need to segment the gang graffiti components from the background. By graffiti components we mean the objects and shapes contained in a graffiti image, such as stars, pitchforks, crowns, and arrows. Gang graffiti components are sprayed in different colors to catch the attention of rival gangs. Therefore, we can use color image segmentation techniques to identify the graffiti components for future analysis.

Since the advent of color imaging most of the image segmentation techniques were proposed for gray-level images [87–90] due to the fact that working with the color channels substantially increases the computational complexity of the method [91]. There has been a remarkable growth on color image segmentation approaches [92–96], which can be divided into three categories [97]: physics based, feature-space based, and image-domain based.

Methods based on physics include dichromatic reflection models [98] and unichromatic reflection models [99] for single illumination sources, and a more general model of image formation [100] for multiple illuminations.

In [98] a method that does not require explicit color segmentation. They separate diffuse and specular reflection components by comparing the intensity logarithmic differentiation of specular-free images and input images iteratively is described. The specular-free image is a set of diffuse components that can be generated by shifting a pixel's intensity and chromaticity nonlinearly while retaining its hue.

Methods based on feature spaces can be sub-categorized into three groups: clustering of regions given patterns with specific properties, including methods such as k-means clustering [101] or Iterative Self-Organizing Data Analysis Technique (ISODATA) [102]; adaptive k-means clustering, including methods based on maximum a posteriori (MAP) estimation [103] or split-and-merge strategies [104]; and histogram thresholding, including methods based on RGB thresholding and hue information [105], specific skin color domains [106], or entropy thresholding [107].

Methods based on the image-domain can be subcategorized into four groups: split-and-merge, including methods such as region smoothing by Markov Random Fields (MRF) [108] or splitting by either watershed transform [109] or quad-tree image representation for segmentation of skin cancers [110], among others; region growing, including methods such as RGB color distribution growing, HSV morphological open-close growing, or color quantization growing [111]; classification based, including methods such as minimization of Hopfield networks [112], or background extraction using two three-layered neural network [113]; edge based techniques, including methods such as combination of HSI gradients [114], active contours, or the Mumford-Shah variation model [115].

In [116, 117] a color histogram for each color channel in the RGB color space is used to detect the most frequently occurring color and segment the background in

food images. Snakes, or active contours, are then used to locate object boundaries and segment images by iteratively minimizing the segmentation energy [118].

In a separate category we can include methods that use external help for segmentation, such as tactile feedback from touchscreens on mobile devices. For example, in [119] a method to extract and segment text from from subway signboard images via touchscreen tracing is presented. The text location is guided by the user selecting the region of interest, and the color information is then used to segment the connected components and use Optical Character Recognition (OCR).

3.1.5 Graffiti Content Analysis

Once the graffiti is segmented from the background we need to analyze its contents. This is done in multiple steps, including image enhancement and reconstruction, straight line removal, and connected component reconnection.

Image enhancement and reconstruction methods can be divided in three categories: spatial filters, neural networks, and fuzzy filters [120].

Spacial filters methods operate directly on the image pixels. In [121] an overview of super-resolution (SR) image reconstruction methods used to increase spatial resolution to overcome the limitations of the sensors and optics is presented. This includes nonuniform interpolation [122, 123], projection onto convex sets (POCS) [124, 125], adaptive filtering [126–128], motionless SR reconstruction [129–131], and blind SR reconstruction [132–134].

Neural network methods try to overcome two main disadvantages of spatial filters: 1) they treat all the pixels in the same way and 2) they operate in single pixels, thus not accounting for characteristics of the neighborhood. In [135] a human visual system (HVS)-directed neural-network-based adaptive interpolation scheme for natural image that produces a higher visual quality for the interpolated image is described. The

pixels pixels of the input image are classified into human perception nonsensitive class and sensitive class, and a neural network interpolates the sensitive regions along edge directions. High-resolution digital images along with supervised learning techniques are used to automatically train the proposed neural network. A supervised method for blood vessel detection and enhancement in digital retinal images is presented in [136]. Vessel enhancement is useful for further extraction of moment invariants-based features. A neural network scheme is used for pixel classification, and a 7-D vector composed of gray-level and moment invariants-based features is used for pixel representation.

Fuzzy filters are less sensitive to local variations and are used when images are corrupted with additive noise [137]. In [138] a method to reduce impulse noise known an "Fuzzy Impulse Noise Detection and Reduction Method (FIDRM)" is described. Based on the concept of fuzzy gradient values, the detection method constructs a fuzzy set impulse noise represented by a membership function that is used by the filtering method, which is a fuzzy averaging of neighboring pixels. The fuzzy set is then used to filter the input image in an iterative fashion. However, FIDRM does not outperform the Median based filters for random impulse noise. In [139] a fingerprint image enhancement method by using fuzzy-based filtering technique and adaptive thresholding is investigated. A process called de-fuzzification, used to produce a quantifiable result in fuzzy logic given fuzzy sets and corresponding membership degrees, is used to improve the contrast of the noisy image.

Straight line removal is the process of deleting lines or segments that do not belong to relevant contents of an image. Since lines have a very similar pattern to character strokes in graffiti images they cannot be eliminated during their initial character extraction stages [17]. Therefore a Hough Transform (HT) is used to detect straight lines in binary images after segmentation and then delete all pixels connected along the lines. After that one reconnects the components originally belonging to graffiti components that intersected with the lines. In [140–142] scratch line detection, re-

moval and restoration on aged films is described. The methods are based on Canny operators, but pixel patches are also used for inpainting [143]. The scratch line detection is based on two general strategies: subdivision of video bands and progressive detection/inpainting. In [144] a method based on energy density and a shear transformation to separate lines from background presented. The shear transform overcomes the disadvantage that linear information loss would happen if the separation method is used only in one direction. Then templates in the horizontal and vertical directions are built to separate lines from background given the fact that the energy concentration of the lines usually reaches a higher level than that of the background in the negative image.

Connected component reconnection is used to merge components that belong to the same object but have been detached during the segmentation or the line removal steps. Contour reconnection methods are widely used in topographic map reconstruction [145]. In [146, 147] the authors propose a method to fill the gaps in contour lines by introducing properties based on geometrical and topological information such as parabolic and opposite directions and differences of y-ordinate of end points. In [148, 149] a method for restoration of degraded digits is presented. The proposed method uses a circular path detection and character stroke analysis based on inertial and centripetal forces. The method then artificially re-creates the stroke segments in order to reconstruct the digit.

3.1.6 Image Features

In order to retrieve similar graffiti images from our database and classify the automatically segmented graffiti components we need to find features that represent images as uniquely as possible. There are four major types of features we can use: color features, texture features, shape features, and local features [150–156]. Given the nature of gang graffiti if we are only interested in describing the segmented graf-

fiti components features like graffiti color or surface texture will not provide useful information. In that case only shape features will be necessary. However, if we want to use information not only from the graffiti components but also from the graffiti background (for image matching and retrieval) we can use color and texture features.

Table 3.1 summarizes some of the state-of-the-art feature types.

Table 3.1: Image feature types and sizes.

Feature	**Type**	**Dimension**	**Notes**
GCH	color	N_C	N_c: Num. colors in quantized space
CCV	color	$2 \times N_C$	
CM	color	$2 \times N_{M_O}$	N_{M_O}: Num. moments
CW-HSV	color	63 bits	
TBD	texture	12 bits	
HTD	texture	$2 \times N_S \times N_K$	N_S: Num. scales, N_K: Num. orientations
EHD	texture	$2 \times N_S \times N_{B_Q}$	N_{B_Q}: Num. borders quantization
Gabor	texture	$2 \times N_S$ (or $2 \times N_K$)	
FD	shape	N_{FD}	N_{FD}: Num. Fourier Descriptors
CSSD	shape	N_P bytes	N_P: Num. peaks on contour map
GMD	shape	N_{M_O}	
ZMD	shape	N_{M_O}	
SIFT	local	128	
SURF	local	64	
PHOW	local	128	
SC	local	$N_\theta \times N_r$	N_θ: Num. angles, N_r: Num. of radius

Color features are the most used visual feature in Content-Based Image Retrieval (CBIR) systems and the most explored features in the literature [157,158]. The main reason is because humans tend to differentiate images mostly by means of color features. The Global Color Histogram (GCH) [159] analyzes the entire color information of the image. Usually, a quantization step is required to reduce the number of distinct colors.

The Color Coherence Vector (CCV) descriptor [160] classifies each pixel in either coherent or incoherent, based on whether or not it is part of a large similarly-colored region. The CCV first blurs the image and the color space is discretized to eliminate

small variations between neighbor pixels. Next, the connected components of the image are found in order to classify the pixels in coherent or incoherent.

The Chromaticity Moment (CM) descriptor [161] characterizes images by chromaticity in the CIE XYZ color space. A chromaticity histogram and a chromaticity trace is generated. The trace indicates the presence of a value (x, y) in the image. The trace and histogram are used to define the chromaticity moments. The reasons for us to choose CM are its compact feature vector generation and its fast distance function, which estimates the modular difference between corresponding moments.

The Color Wavelet HSV (CW-HSV) descriptor [162] computes color features in the wavelet domain [163]. First the image global color histogram in HSV color space is found. Then the Haar transform coefficients of the histogram are determined hierarchically by using Haar wavelet functions. In the end, 63 binary values compose the feature vector. The distance between two feature vectors is calculated by the Hamming distance. The reasons for us to choose CW-HSV are its compact feature vector generation (only 63 bits) and its fast distance function.

Texture features, like color features, create powerful low-level descriptors for image search and retrieval applications [164].

The Texture Browsing Descriptor (TBD) [165] relates to the perceptual characterization of texture, in terms of regularity, directionality and coarseness. The coarseness is related to image scale or resolution. This descriptor is useful for browsing type applications and coarse classification of textures. The Homogeneous Texture Descriptor (HTD) [164] provides a quantitative characterization of homogeneous texture regions for similarity retrieval. It is determined by first filtering the image with a bank of orientation and scale sensitive filters, and computing the mean and standard deviation of the filtered outputs in the frequency domain.

The local Edge Histogram Descriptor (EHD) [164] is useful when the underlying region is not homogeneous in texture properties. It is computed by first sub-dividing the image and computing local edge histograms. Edges are broadly grouped into five

categories: vertical, horizontal, 45 diagonal, 135 diagonal, and isotropic. Thus, each local histogram has five bins, and with the image partitioned into 16 sub-images results in 80 bins. The Gabor-based descriptor [166] is computed by passing the image through a bank of Gabor filters [167]. Filters in a Gabor filter bank can be considered as edge detectors with tunable orientation and scale so that information on texture can be derived from statistics of the outputs of those filters. The descriptor is then formed as a vector of means and standard deviations of filter responses.

Shape features are one of the primary low level image features exploited in content-based image retrieval [168]. They can represent images by their contours or regions.

The Fourier Descriptor (FD) [169–171] is a spectral descriptor obtained from a Fourier transform on a shape signature. The shape signature is a one-dimensional function, which is derived from shape boundary coordinates. The set of normalized Fourier transformed coefficients is known as the Fourier descriptor of the shape. The Curvature Scale Space Descriptor (CSSD) [172, 173] treats shape boundary as a 1D signal, and analyzes this 1D signal in scale space. By examining zero crossings of curvature at different scales, the concavities/convexities of shape contour are found. These concavities/convexities are useful for shape description because they represent the perceptual features of shape contour.

The Geometric Moment Descriptor (GMD) [174, 175] is based on moment invariants for shape representation and similarity measure. Moment invariants are derived from moments of shapes, and are invariant to 2D geometric transformations of shapes. The Zernike Moment Descriptor (ZMD) [176,177] uses orthogonal moments to recover the image from moments based on the theory of orthogonal polynomials (Zernike polynomials). It allows independent moment invariants to be constructed to an arbitrarily high order.

Local features rely on the concept that objects in images consist of parts that can be modeled with varying degrees of independence [178, 179]. They are used in many applications, such as object detection, symbol spotting, or image registration.

The Scale Invariant Feature Transform (SIFT) descriptor [15] combines a scale invariant region detector and a descriptor based on the gradient distribution in the detected regions. The descriptor is represented by a 3 dimensional histogram of gradient locations and orientations. The contribution to the location and orientation bins is weighted by the gradient magnitude. The quantization of gradient locations and orientations makes the descriptor robust to small geometric distortions and small errors in the region detection. The Speed Up Robust Feature (SURF) descriptor [24] is based on similar properties as SIFT, but relies on integral images for image convolutions. First, it fixes a reproducible orientation based on information from a circular region around the interest point. Then, it constructs a square region aligned to the selected orientation, and extract the SURF descriptor from it.

The Pyramid Histogram Of visual Words (PHOW) descriptor [180, 181] is computed using SIFT on a dense gird at a fixed scale, which can be directly clustered using k-means [182] to form a "bag of words" feature. The Shape Context (SC) descriptor [183–185] is similar to the SIFT descriptor, but is based on edges. It is a 3 dimensional histogram of edge point locations and orientations. The edge locations are quantized into a log-polar coordinate system and the orientations are quantized into an angular coordinate system.

3.1.7 Image Retrieval

Retrieval of gang graffiti images is very useful for the first responder in the field. It can provide information about related graffiti in the area based on the contents of the image. For example, a user can check if someone else has taken an image of the same gang graffiti in the past, and pull all the related information without having to

do any further image analysis.

Content-Based Image Retrieval (CBIR) can be used for finding images from large and unannotated image databases. There are four core techniques for CBIR: visual signature, similarity measures, classification and clustering, and search paradigms [186–188]. Visual signature usually involves three steps: 1) segmenting images using methods such as k-means clustering [182], normalized cuts [189], or salient region detection [190]; 2) using features such as color, texture, or shape [191]; 3) constructing the signatures (or feature vectors) using distributions [192] or adaptivity [193]. Similarity measure methods include manifold embedding [194], and vector quantization [195]. Classification and clustering methods include hierarchical k-means [196], support vector machine [197], or Bayesian classifiers [198]. Search paradigms methods include learning-based [199], probabilistic [200], region-based [201], feedback specification [202], or user-driven [203].

In [204] a method for image-based retrieval using a mobile device is presented. Features are measured after detecting salient regions and then quantified to form a vector using a clustering-based bag-of-words model and sparse matrix methods. Invert document methods are used to speed up real-time queries. In [11] a CBIR system tattoo image retrieval is proposed. The system automatically uses SIFT features and additional information (i.e., body location of tattoos and tattoo classes) to improve the retrieval time and retrieval accuracy. Geometrical constraints are also introduced in SIFT keypoint matching to reduce false retrievals.

Sketch-Based Image Retrieval (SBIR) uses a line-based hand-drawing (a "sketch") as a query. In some scenarios outline sketches are typically easier and faster to generate than a complete color description of the scene [205, 206].

In [207] a method based on elastic matching of sketched templates over the shapes in the images to evaluate similarity ranks is described. The degree of matching achieved and the elastic deformation energy spent by the sketch to achieve such a

match are used to derive a measure of similarity between the sketch and the images in the database and to rank images to be displayed. The elastic matching is integrated with arrangements to provide scale invariance and take into account spatial relationships between objects in multi-object queries.

In [208] a technique that deals with images containing several complex objects in an inhomogeneous background is presented. Two abstract images are obtained using strong edges of the model image and the morphologically thinned outline of the sketched image. The angular-spatial distribution of pixels in the abstract images is then employed to extract new compact and effective features using the Fourier transform. The features are rotation and scale invariant and robust against translation.

The image retrieval method used in GARI fall into the feature-space category in CBIR. However, our approach differs from the methods mentioned above. Although there are some techniques in the literature that use only hue or luma information, either circular histogram thresholding [209] or one-dimensional histogram thresholding [210], we do not obtain the descriptors of the probability distribution from the color histogram of the image. Instead, the median and the variance obtained from the tracing-bases color recognition process are used for segmentation. Our segmentation approach does not produce binarized images, but grayscale images weighed by a Gaussian distribution, thus creating a probability map for a specific luma or hue. These types of probability maps are used for increased accuracy and robustness in some clustering techniques [211, 212]. Our content based image retrieval approach uses hierarchical k-means to build a vocabulary tree based on the method in [196].

3.2 Mobile-Based Motion Blur Prevention and Detection

In order to analyze gang graffiti we need to preserve the details in the image acquired with a mobile device. Instead of doing blur detection after taking the image we propose a mobile-based method to prevent the user from producing blurred im-

ages. To that end we use a customized camera function on the mobile that detects shake events (i.e. motion blur). When the camera function is launched through the GARI application we start a three second countdown and listen for changes from the accelerometer sensor in the mobile device. A sensor of this type measures the acceleration of the device (Ad) in SI units (m/s^2). Conceptually, this is done by measuring forces applied to the sensor itself (Fs) using the relation:

$$Ad = -\frac{\sum Fs}{mass}. \tag{3.1}$$

In particular, the force of gravity is always influencing the measured acceleration:

$$Ad = -g - \frac{\sum Fs}{mass}. \tag{3.2}$$

For this reason when the device is sitting on a table the accelerometer reads a magnitude of $g = 9.81m/s^2$. Similarly, when the device is in free-fall its accelerometer reads a magnitude of $0m/s^2$. We compute the total movement M as

$$M = \Delta A_x + \Delta A_y + \Delta A_z - (A_x + A_x + A_z), \tag{3.3}$$

where (Δ_x, Δ_y, Δ_z) are the acceleration force changes along the (x, y, z) axes respectively, and (A_x, A_y, A_z) are the most recent acceleration values along the (x, y, z) axes respectively. If ΔA and A occur in a time difference of $T_t = 400$ milliseconds and M is above a threshold $T_M = 3m/s^2$ we report a shake event. In that case the countdown is reset to three seconds and no image is taken. If no significant change on M is perceived when the countdown reaches zero, we trigger the auto-focus and an image is acquired.

Even though we try to prevent motion blur, if there is a shake event during auto-focus or image acquisition we can obtain a blurred image. For this reason motion

blur detection is done on a reduced size version of the image of width $W_t = 400$ pixels. A reduced size version is enough to detect excessive motion blur produced in this particular case. We use a modification of the method proposed in [44] because of its simplicity and speed. In [44] a modification to a well known method known as cumulative probability of blur detection (CPBD) is presented. This utilizes the probability distribution of edge widths [51]. The blur metric estimation starts by creating an edge binary map using a Sobel operator in the vertical direction of the grayscale image. Then, the image is divided into blocks of size 64×64. A block is considered an edge block of it contains a number of edge pixels greater than a fixed threshold. For each edge block the probability of blur detection P_{BLUR} at each edge pixel e_i is computed as

$$P_{BLUR}(e_i) = 1 - e^{-\left|\frac{w(e_i)}{w_{JNB}(e_i)}\right|^{\beta}}, \tag{3.4}$$

where $w(e_i)$ is the edge width [49], $w_{JNB}(e_i)$ is the "just noticeable blur" (JNB) width with value of either 5 or 3 [51], and β is a parameter whose value is obtained from least squares fitting. The CPBD is estimated as:

$$CPBD = P(P_{BLUR} \leq P_{JNB}) = \sum_{P_{BLUR}=0}^{P_{BLUR}=P_{JNB}} P(P_{BLUR}), \tag{3.5}$$

where $P(P_{BLUR})$ denotes the value of the probability distribution function at a given P_{BLUR}. This metric is based on the fact that, at the JNB, $w(e_i) = w_{JNB}(e_i)$, which corresponds to the probability of blur detection $P_{BLUR} = P_{JNB} = 63\%$. Therefore, for a given edge e_i, when $P_{BLUR} \leq P_{JNB}$ the edge is considered not to be blurred. Hence, a higher metric value represents a sharper image. The modification proposed by [44] relies on the fact that the CPBD can be expressed by the ratio

$$CPBD = P(P_{BLUR} \leq P_{JNB}) = \frac{|S_1|}{|S_e|}, \tag{3.6}$$

where $|S_1|$ is the set of edge pixels with $P_{BLUR} \leq P_{JNB}$ and $|S_e|$ is the set of all edge pixels. Since

$$1 - e^{-\left|\frac{w(e_i)}{w_{JNB}(e_i)}\right|^{\beta}} \leq 0.63 \Rightarrow w(e_i) \leq w_{JNB}(e_i)(-ln(0.37))^{1/\beta} \tag{3.7}$$

the CPBD becomes

$$CPBD = \frac{\sum_{w_{JNB}=\{3,5\}} \sum_{w=2}^{w_{JNB}-1} H(w_{JNB}, w)}{|S_e|}, \tag{3.8}$$

where $H(w_{JNB}, w)$ is the number of edge pixels with JNB width w_{JNB} and edge width w. By using this approach we avoid using exponentials for gradient estimations, thus reducing the computational complexity.

We can further increase the complexity by approximating the CPBD as

$$BM = \frac{\sum_{x,y} |G_x(x,y)| + \sum_{x,y} |G_y(x,y)|}{w_I h_I}, \tag{3.9}$$

where (G_x, G_y) are the Sobel derivatives in the x and y directions respectively, and (w_I, h_I) are the dimensions of the image. That is, BM is the ratio of edge pixels over the size of the image. Note that by doing this we cannot call the metric CPBD, since it is not based on cumulative probability.

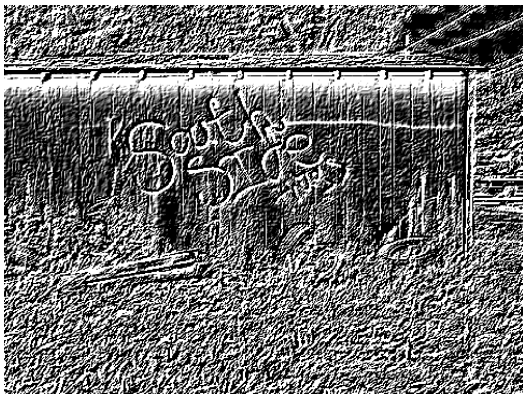
By using BM as our blur metric we can set a threshold T_{BM} so that if $BM < T_{BM}$ we consider the image to be blurred, and we ask the user to retake the image by reseting the countdown back to three. Our experiments with more than 1,000 images

from our dataset and different mobile devices showed that $T_{BM} = 0.1$ produces the best results.

Note that since the proposed blur detection metric is solely based on the number of edge pixels, the method will also reject images with large uniform patches or images taken under low light conditions. In fact this properties are not a drawback, but rather desired in the context of gang graffiti recognition and interpretation. Also note that all the processing is done on the mobile device.

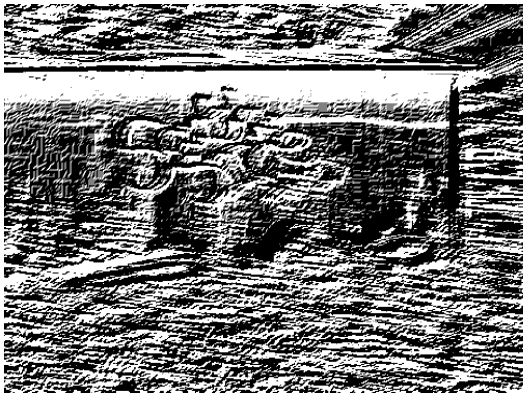
Figure 3.1 shows an example of the blur metric results.

(a) Original Image



(b) Sobel Gradients. BM = 0.21

(c) Original Image



(d) Sobel Gradients. BM = 0.07

Fig. 3.1.: Example of Blur Metric Results.

Table 3.2 shows all the parameters/thresholds we used including empirically derived parameters.

Table 3.2: Parameters and thresholds used in Mobile-Based Motion Blur Prevention.

Parameter	**Description**	**Value**
T_t	Time between acceleration changes	400 ms
T_M	Threshold to consider shake event	$3m/s^2$
W_t	Width of resized image for blur detection	400 px
T_{BM}	Threshold for Blur Metric (BM) ratio	0.1

3.3 Color Correction Based on Mobile Light Sensor[1]

First responders are out in the field when using the mobile application to take images of gang graffiti. Since gang graffiti are usually found in dangerous neighborhoods we want to minimize the use of intrusive methods to do color correction. The use of fiducial markers may be suspicious to gang members in the surroundings. The use of face detection for white balancing [213] make first responders concerned about their privacy.

One way to do color correction is to first obtain information about the scene illumination. This can be done by using the light sensor on the mobile device. For example, the light sensor in an Android smartphone returns the ambient light level in SI lux units (lumens per square meter). Unlike human perception of light, lux readings are directly proportional to the energy per square meter that is absorbed per second [214,215]. However, human perception can be simplified by creating several ranges of interest with known upper and lower thresholds. Table 3.3 shows an example of several thresholds for common lighting conditions and the corresponding lighting steps obtained from the light sensor on a Samsung Galaxy Nexus smartphone. Each lighting step represents a change in lighting environment. Figure 3.2 illustrates the relationship between the lighting step and the lux values. Figure 3.3 illustrates the same relationship when using a logarithmic scale on the lux values to see how the relationship becomes linear.

Once we obtain a lux LX from the mobile device we want to associate a color correction matrix to it. A color correction matrix is a mapping between and image illuminated with reference lighting and and image acquired with unknown lighting condition.

The idea is to generate color correction matrices from ground-truth data to populate a database. The database acts as a look up table where a lux value maps to a color correction matrix. Later, when first responders use the application in the field

[1]The work presented in this section is partly based on the work by my Purdue colleague Dr. Chang Xu [44].

Table 3.3: Thresholds for common lighting conditions and corresponding lighting steps.

Condition	**Lux (start)**	**Lux (end)**	**Lighting step**
Pitch Black	0	10	1
Very Dark	11	50	2
Dark Indoors	51	200	3
Dim Indoors	201	400	4
Normal Indoors	401	1000	5
Bright Indoors	1001	5000	6
Dim Outdoors	5001	10,000	7
Cloudy Outdoors	10,001	30,000	8
Direct Sunlight	30,001	100,000	9

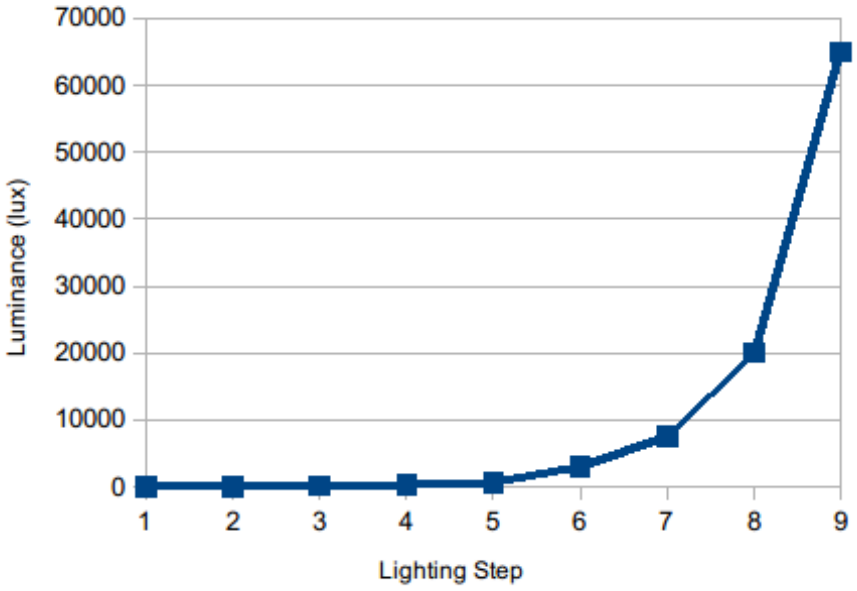


Fig. 3.2.: Lighting Step vs. Luminance (lux).

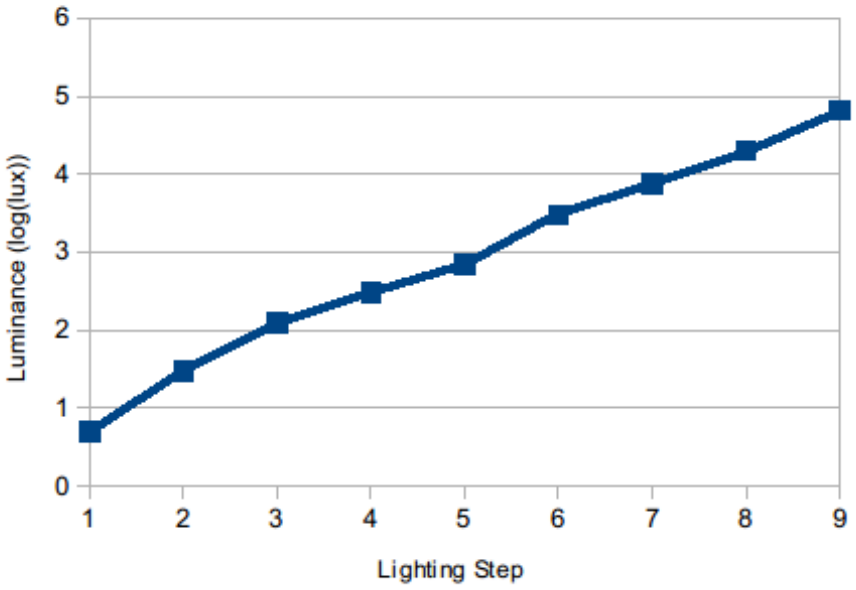


Fig. 3.3.: Lighting Step vs. Luminance (log(lux)).

we will only need a lux value to retrieve the corresponding color correction matrix and use it to correct the acquired image.

Figure 3.4 illustrates the process to populate the database with color correction matrices and lux values. Note that the computation of the color correction matrix $M_{GT \rightarrow D65}$ is done on the mobile device. A ground-truth image is an image acquired with a mobile device under a specific scene illumination. Figure 3.5 shows an example of a ground-truth image with a lux value of 5,116. The image contains a checkerboard-like design known as a "fiducial marker" used as a reference of known dimensions and color patches [44, 61, 216].

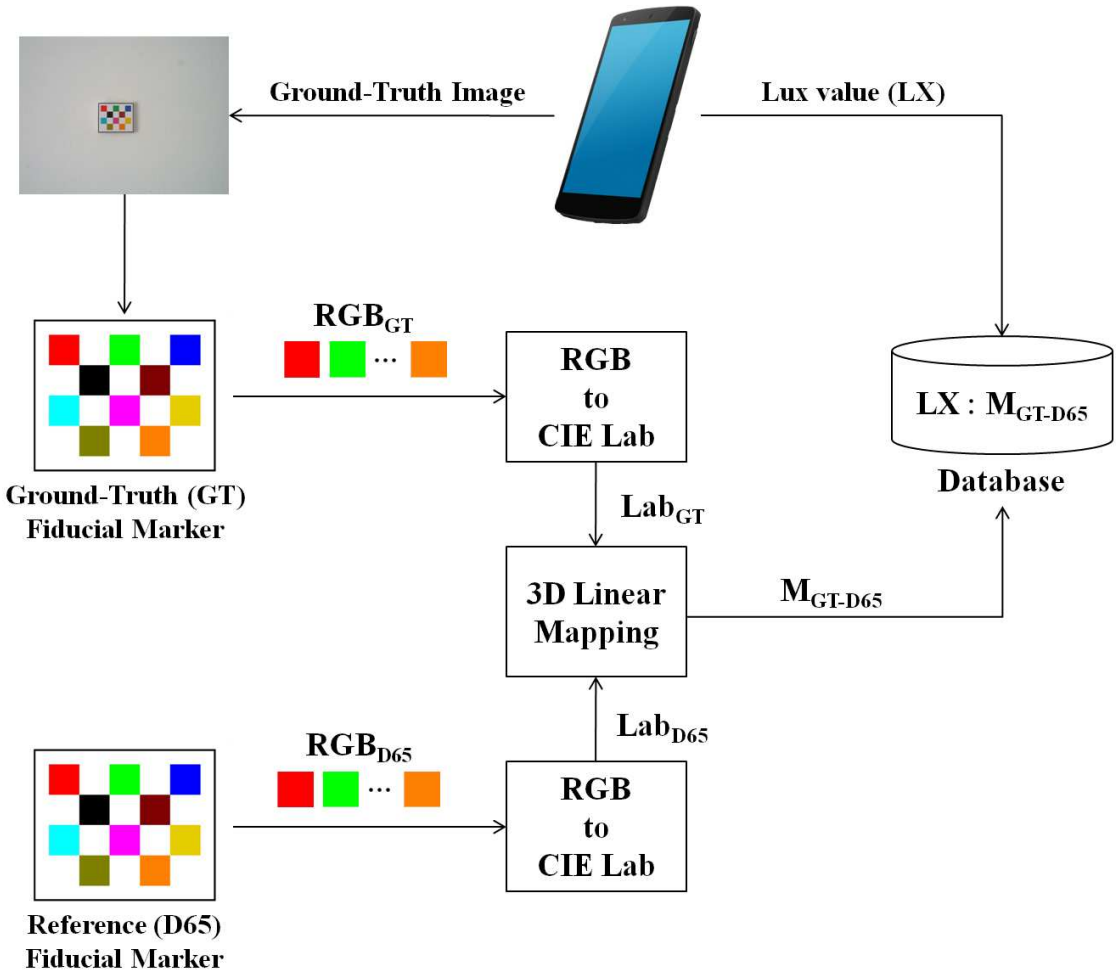


Fig. 3.4.: Color Correction Based on Mobile Light Sensor.

Fig. 3.5.: Example of ground-truth image with a lux value of 5,116.

We start by detecting the corners of the fiducial marker in the ground-truth image using the method described in [217]. The image is first converted to grayscale and binarized according to

$$I_{out}(x,y) = \begin{cases} 255 & \text{if } I(x,y) > T(x,y) \\ 0 & \text{else} \end{cases}, \tag{3.10}$$

where $T(x,y)$ is a threshold calculated individually for each pixel using a Gaussian kernel. The kernel is a matrix of Gaussian filter coefficients:

$$G_i = \alpha \exp^{-\frac{i-\left(\frac{k-1}{2}\right)^2}{(2\sigma)^2}}, \tag{3.11}$$

where k is the aperture size (odd and positive), σ is the Gaussian standard deviation computed as $\sigma = 0.3((k-1)^{1/2}-1)+0.8$, $i = 0, ..., k-1$ and α is the scale factor chosen so that $\sum_i G_i = 1$. The binary image is eroded to separate the checkerboard at the corners and obtain a set of quadrangles. Finally, a quadrangle linking step checks the position of the fiducial marker patches to confirm the board pattern.

Once we have detected the location of the checkerboard corners we estimate the location of each of the 11 color patches and extract their mean RGB value. These color patches are used to generate a 3D linear mapping between the scene illumination (ground-truth image) and the reference fiducial marker colors [218]. We used the linear model in LAB color space from [44] for color correction, as it produced the best results in our experiments (Section 5.1.2). We convert each of the RGB color patches to CIE Lab using the standard RGB to CIE Lab transformation [219, 220] as follows:

RGB to XYZ:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{3.12}$$

XYZ to CIE Lab:

$$L = 116(Y/Y_n)^{1/3} - 16 \tag{3.13}$$

$$a = 500((X/X_n)^{1/3} - (Y/Y_n)^{1/3}) \tag{3.14}$$

$$b = 200((Y/Y_n)^{1/3} - (Z/Z_n)^{1/3}) \tag{3.15}$$

X_n, Y_n and Z_n are the values of X, Y and Z for the illuminant (reference white point). The L coordinate in CIE Lab is correlated to perceived lightness. The a and b coordinates are the red-green and yellow-blue of the color-opponent respectively. We followed the ITU-R Recommendation BT.709 and used illuminant D_{65}, where $[X_n,\ Y_n,\ Z_n] = [0.950456,\ 1,\ 1.088754]$ [221]. To obtain the optimal 3-dimensional linear transformation $M_{GT \to D65}$, a 3×3 matrix that converts the Lab color patches from the ground-truth to the Lab color patches from the D_{65} reference, we need to solve

$$M_{GT \to D65} = \underset{M_{3\times3}}{\operatorname{argmin}} \sum_{i=1}^{11} \left\| (Lab_i)^T_{D65} - M_{3\times3}(Lab_i)^T_{GT} \right\| \tag{3.16}$$

by linear regression by using ordinary least-squares estimates of the regression coefficients [222]. We follow this procedure for each ground-truth image to populate the database with mappings between lux values LX and color correction matrices $M_{GT \to D65}$.

Every time a user acquires an image I_q using the mobile device we sent it to the server along with the lux value LX_q. Then, we use the $M_{GT \to D65}$ associated to the closest LX in the database to correct I_q.

Figures 3.6 and 3.7 show example outputs of our proposed color correction method. Details about the number of ground-truth images used and the efficiency of the method are described in Section 5.1.2.

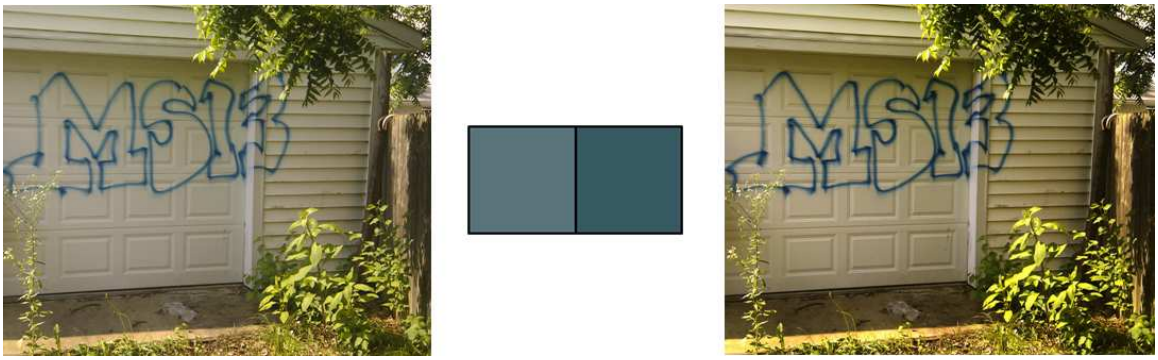


Fig. 3.6.: Example of color correction when $LX = 35,611$. Left: before correction; right: after correction.

Fig. 3.7.: Example of color correction when $LX = 41,980$. Left: before correction; right: after correction.

3.4 Color Recognition Based on Touchscreen Tracing

In this method the user acquires an image of a gang graffiti and traces a path along a colored region using the touchscreen display. Then we recognize the color along the path and provide a list of gangs related to the color by querying an internal database on the mobile phone. For this method we use an RGB to Y'CH color space conversion. Figure 3.8 shows an overview of our color recognition method. Again note that this technique is done on the hand-held device.

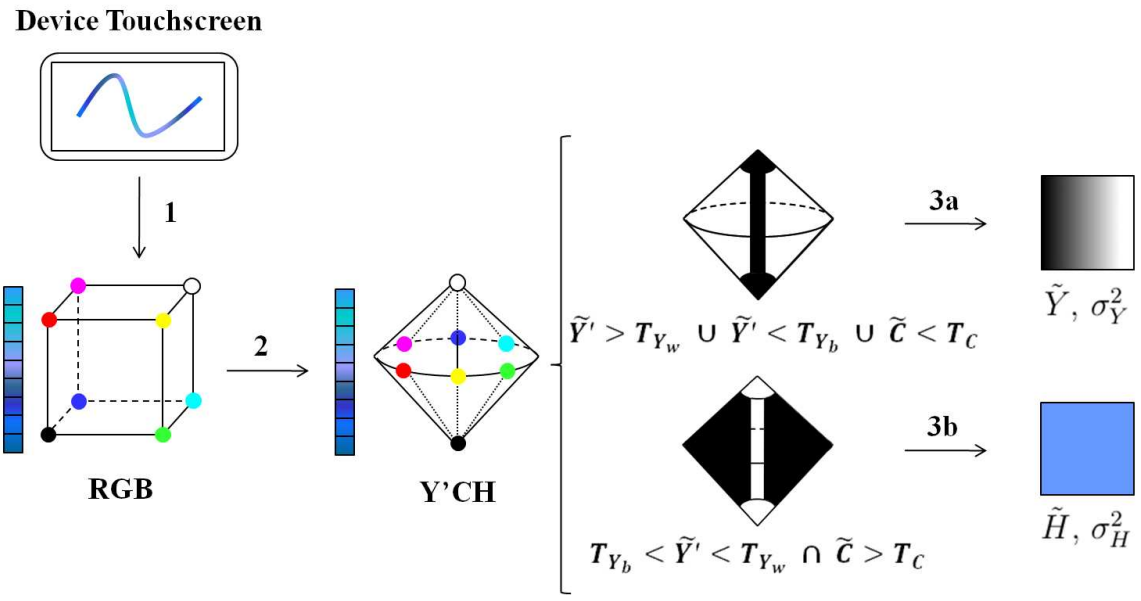


Fig. 3.8.: Color Recognition Based on Touch Screen Tracing.

First, the user captures an image or browses the internal gallery for an image on the device and draws a path with the finger on the touchscreen. The path is drawn along a graffiti component on the image assumed to be sprayed in uniform color. The RGB color channels of each pixel on the path are converted to a new luma/chroma/hue color space that we call the Y'CH color space. The Y'CH color space is used because color changes are more intuitive and perceptually relevant to represent in luma or hue than in RGB triplets, in order to obtain the median and the variance of the color along the traced path. Equation 3.17 shows the mapping between RGB and

Y'CH. Note that we use luma (Y') as opposed to luminance (Y) [223]. Appendix A describes in detail the RGB to Y'CH color space conversion using both an arithmetic approach and a trigonometric approach. We compute three medians on the pixel array that forms the path, namely the luma median ($\tilde{Y}$), the chroma median ($\tilde{C}$) and the hue median ($\tilde{H}$). We then define two disjoint regions in our Y'CH color space (luma region and hue region, labeled 3a and 3b in Figure 3.8 respectively), delimited by manually set thresholds based on luma ($T_{Y_w} = 0.12$, $T_{Y_b} = 0.85$) and chroma ($T_C = 0.06$). These thresholds were empirically obtained from our database of gang graffiti, consisting of more than 700 gang graffiti images. Depending on the region where the medians are located, we do color recognition based on luma (3a) or hue (3b).

$$
\begin{aligned}
&Y' = 0.299R + 0.587G + 0.114B. \\
&C = max(R,G,B) - min(R,G,B) = M - m \\
&H = \begin{cases} \frac{G-B}{C} & \text{if } M = R \\ \frac{B-R}{C} + 2 & \text{if } M = G \\ \frac{R-G}{C} + 4 & \text{if } M = B \\ 0 & \text{if } C = 0 \end{cases}
\end{aligned} \tag{3.17}
$$

Once we have the median, either based on luma or hue, we need to decide which color is associated with it. From all the images in our database, the possible colors used on graffiti are black, white, red, blue, green, gold and purple. If the median is based on luma, the color detected is either black ($\tilde{Y} \leq T_{\tilde{Y}}$) or white ($\tilde{Y} > T_{\tilde{Y}}$), where $T_{\tilde{Y}} = 0.5$. If the median is based on hue, the color detected is $H_d = \min_i(\theta(\tilde{H}, H_{A_i}))$, where $\theta(\tilde{H}, H_{A_i})$ is the angular distance between the computed hue ($\tilde{H}$) and the i-th component of a set of average hues (H_A), empirically obtained from analyzing 100 color calibrated images taken from our database. These colors are specified in Table 3.4. Figure 3.9 illustrates the separation between them in a hue slice of the Y'CH

color space. Once the color is detected, we provide a list of gangs related to that color by querying our database of gang graffiti from the mobile phone.

Finally, we also estimate the variance $\sigma^2_{\tilde{X}}$ near the median $\tilde{X} = \{\tilde{Y} \text{ or } \tilde{H}\}$. This variance is used as an input to the color image segmentation method described next. Note that this method can be used with multi-colored graffiti by using it on each trace on the touchscreen.

Table 3.4 shows all the parameters/thresholds we used including empirically derived parameters.

Table 3.4: Parameters and thresholds used in Color Recognition Based on Touchscreen Tracing.

Parameter	**Description**	**Value**
T_{Y_w}	Low luma threshold	0.12
T_{Y_b}	High luma threshold	0.85
T_C	Low chroma threshold	0.05
$T_{\tilde{Y}}$	Luma threshold for black/white	0.5
H_A^{red}	Average hue (red)	6.10 rad
H_A^{blue}	Average hue (blue)	4.00 rad
H_A^{green}	Average hue (green)	2.20 rad
H_A^{gold}	Average hue (gold)	0.69 rad
H_A^{purple}	Average hue (purple)	5.15 rad

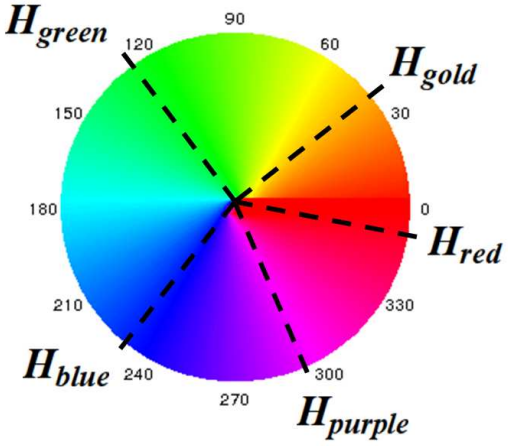


Fig. 3.9.: Separation Between Hue Averages.

3.5 Automatic Graffiti Component Segmentation

In this section we propose methods for automatic segmentation of graffiti components. We assume that the graffiti takes at least 50% of the image. With this assumption we resize all input images to $W_X = 500$ pixels in width to reduce the computational complexity while maintaining the performance.

3.5.1 Color Image Segmentation Based on Gaussian Thresholding

For the segmentation we use a Gaussian threshold near a specific luma or hue value in the Y'CH color space, in order to produce a segmented image where each pixel is given a weight depending on its distance from a median. Figure 3.10 shows an overview of our color segmentation method divided in 5 steps. Note that we currently use this method on the server in our system and do not use it on the hand-held device.

We assume that, given a graffiti image X, we have the median $\tilde{X}$ and the variance, $\sigma^2_{\tilde{X}}$, of a traced path (step 1b). We then transform the entire RGB image to the our Y'CH color space (steps 1a and 2). Finally, we segment the image using Gaussian thresholding (steps 3 to 5). The segmentation works as follows. We first ignore all pixels in the image X that fall outside the region established during touchscreen tracing (luma or hue), using the same thresholds used for the color recognition process. This creates the thresholded grayscale image X_t (step 3). We weight the rest of the pixels using a normal distribution centered at $\tilde{X}$ and a confidence interval of $2\sigma_{\tilde{X}}$ (step 4), as shown in Equation 3.18, to obtain X_g. The output X_g is a grayscale image where each pixel is given a probability based on a normal distribution (step 5). This probability is higher as the pixel value gets closer to $\tilde{X}$. The image is then scaled to $[0, 255]$.

$$X_g(i,j) = \begin{cases} \frac{1}{\sqrt{2\pi\sigma^2_{\tilde{X}}}} e^{-\frac{(X_t(i,j)-\tilde{X})^2}{2\sigma^2_{\tilde{X}}}} & |X_t(i,j)| < 2\sigma_{\tilde{X}} \\ 0 & \text{else} \end{cases} \tag{3.18}$$

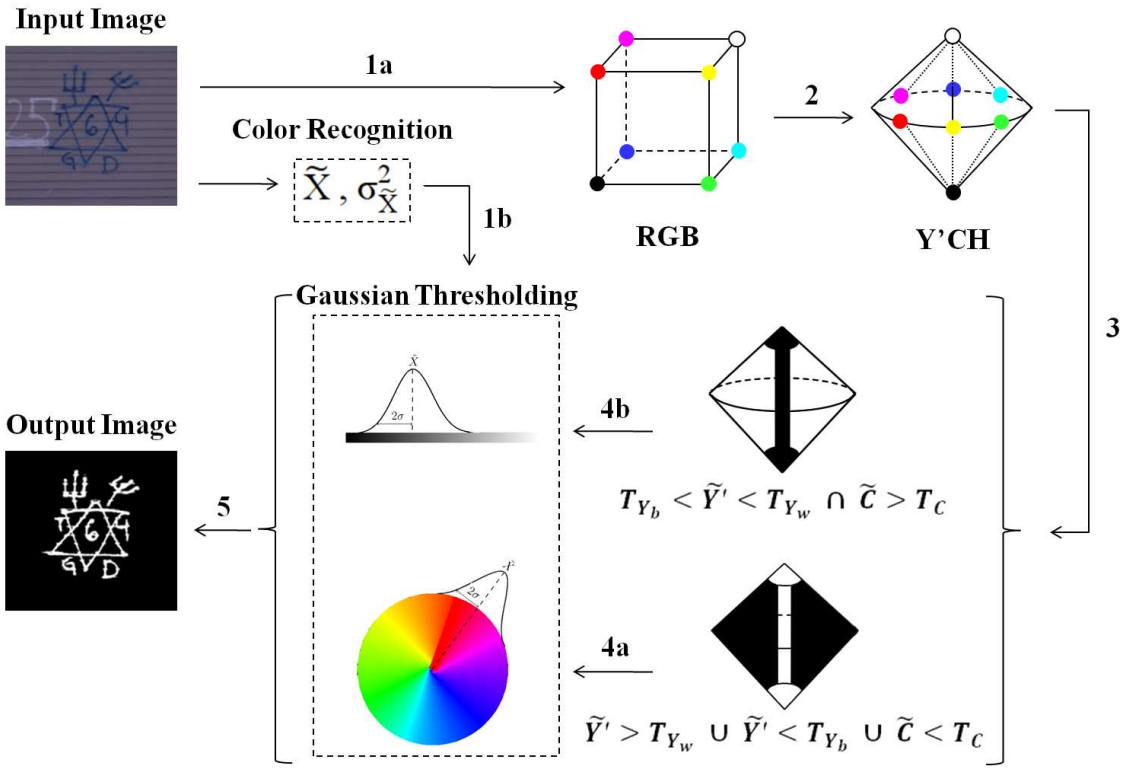


Fig. 3.10.: Color Image Segmentation Using Gaussian Thresholding.

Figure 3.11 shows an example where the color recognition is done by tracing a path along the blue numbers "2" and "5". Figure 3.12 shows the effect of the Gaussian thresholding process on the letters "Hill". Note that this method produces a probability map, where the values in a graffiti component decrease as the spray paint fades. This indicates how the graffiti was traced, and it may be useful in future research for shape analysis (Section 6). Appendix B illustrates more examples of our color segmentation method.

(a) Input Image (b) Segmented Image

Fig. 3.11.: Gaussian Thresholding on Blue. $(\tilde{H}, \sigma^2_{\tilde{H}}) = (4.19, 0.05)$.

Table 3.5 shows all the parameters/thresholds we used including empirically derived parameters.

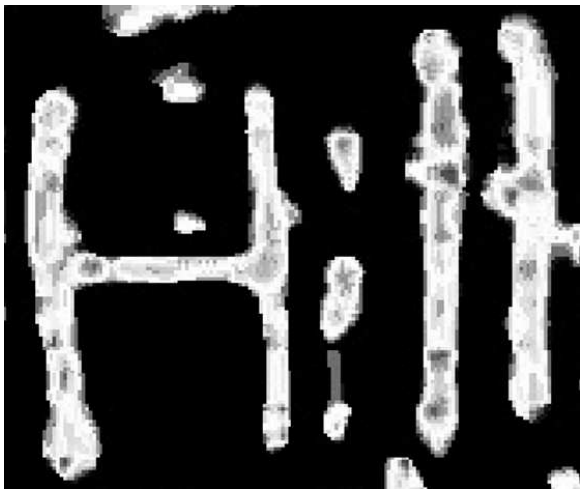


Fig. 3.12.: Probability Map Created By The Gaussian Thresholding.

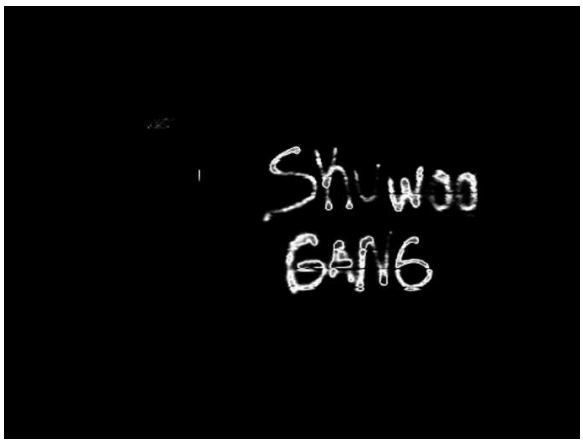
Table 3.5: Parameters and thresholds used in Color Image Segmentation Based on Gaussian Thresholding. W_X and H_X are the width and height of X respectively.

Parameter	**Description**	**Value**
W_X	Width of resized image for image segmentation	500 px
T_{Y_w}	Low luma threshold	0.12
T_{Y_b}	High luma threshold	0.85
T_C	Low chroma threshold	0.05

3.5.2 Block-Wise Gaussian Segmentation Enhancement

Since the median and variance for Gaussian thresholding are obtained from a small sample of the graffiti the resulting probability map X_g can contain broken or fainted graffiti components and noise. These can be caused by either non-uniform scene illumination (Figure 3.13) or foreground-background hue similarity (Figure 3.14).

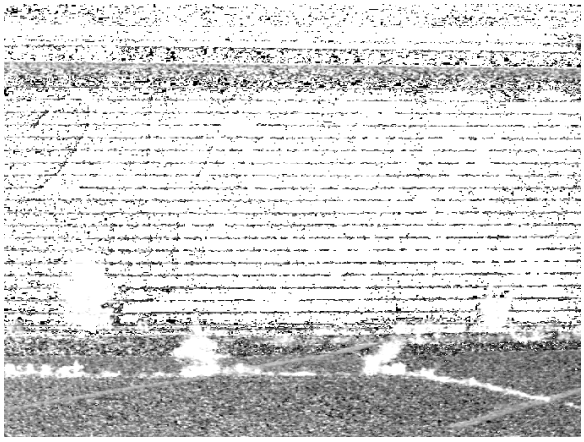
(a) Original Image. The traced path is marked in green.



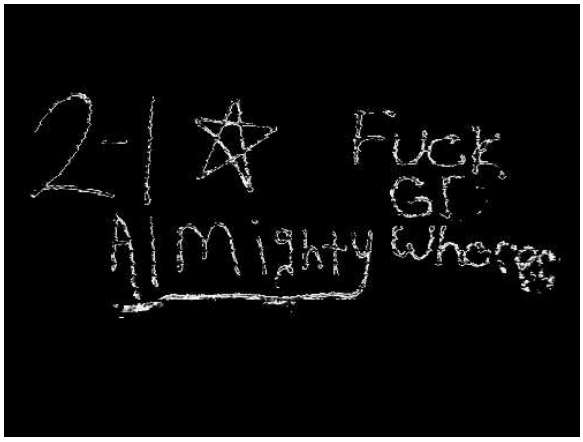
(b) Gaussian Thresholding

Fig. 3.13.: Gaussian Thresholding results with non-uniform scene illumination.

(a) Original Image. The traced path is marked in green.



(b) Hue Channel



(c) Gaussian Thresholding

Fig. 3.14.: Gaussian Thresholding results with foreground-background hue similarity.

Therefore, we need to enhance X_g before finding the graffiti components. This can be done by using a block-wise median filter on the luma, chroma and hue channels of the original image X separately and merging the results. Figure 3.15 shows the process.

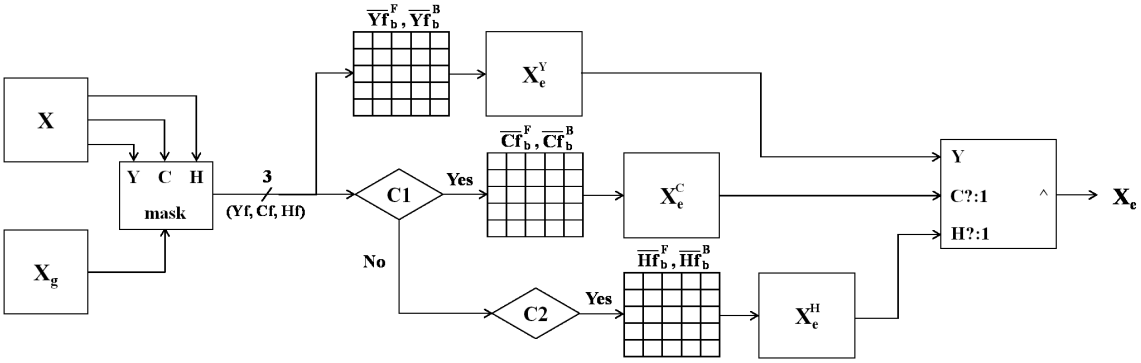


Fig. 3.15.: Block-Wise Gaussian Segmentation Enhancement.

First, we filter each channel on X with a binary mask created from X_g, so that

$$Yf(x,y) = \begin{cases} Y(x,y) & \text{if } X_g(x,y) > 0 \\ 0 & \text{else} \end{cases} \tag{3.19}$$

$$Cf(x,y) = \begin{cases} C(x,y) & \text{if } X_g(x,y) > 0 \\ 0 & \text{else} \end{cases} \tag{3.20}$$

$$Hf(x,y) = \begin{cases} H(x,y) & \text{if } X_g(x,y) > 0 \\ 0 & \text{else} \end{cases} \tag{3.21}$$

Then, we divide Yf in blocks of size $w_s \times w_s$, where $w_s = 0.03max(W_X, H_X)$ and (W_X, H_X) are the width and height of X respectively. We only consider blocks $b \in B$, where B is the set of blocks containing at least one non-zero valued pixel. For each block $b \in B$ we compute the luma median of the foreground pixels $\widetilde{Yf}_b^F$ and the luma

median of the background pixels $\widetilde{Yf}_b^B$. Then, we generate the binary image X_e^Y by evaluating each individual pixel:

$$X_e^Y(x,y) = \begin{cases} 1 & \text{if } b \in B \text{ and } |Y(x,y) - \widetilde{Yf}_b^F| < |Y(x,y) - \widetilde{Yf}_b^B| \\ 0 & \text{else} \end{cases}, \quad (3.22)$$

where b is the block associated with the coordinates (x,y). We use the chroma channel for enhancement if $\frac{\sum_{b\in B}|\widetilde{Cf}_b^F - \widetilde{Cf}_b^B|}{n(B)} > T_e^C$ (condition C1 in Figure 3.15), where $n(B)$ is the cardinality of B. A value of $T_e^C = 0.06$ produced the best results after running experiments on more than 700 gang graffiti images. In that case,

$$X_e^C(x,y) = \begin{cases} 1 & \text{if } b \in B \text{ and } |C(x,y) - \widetilde{Cf}_b^F| < |C(x,y) - \widetilde{Cf}_b^B| \\ 0 & \text{else} \end{cases}, \quad (3.23)$$

If $\frac{\sum_{b\in B}|\widetilde{Cf}_b^F - \widetilde{Cf}_b^B|}{n(B)} \leq T_e^C$ we can still use the hue channel for enhancement. If X_g was obtained using the hue channel during the Gaussian Thresholding (i.e. $\widetilde{X} = \widetilde{H}$) (condition C2 in Figure 3.15) we apply an additional threshold to each pixel. In this case we keep pixels where the hue angular distances satisfy $\theta(H(x,y), \widetilde{Hf}_b^F) < \theta(H(x,y), \widetilde{Hf}_b^B)$, where $\theta(a,b) = |mod(a-b, 2\pi) - \pi|$. That is,

$$X_e^H(x,y) = \begin{cases} 1 & \text{if } b \in B \text{ and } \theta(H(x,y), \widetilde{Hf}_b^F) < \theta(H(x,y), \widetilde{Hf}_b^B) \\ 0 & \text{else} \end{cases}. \quad (3.24)$$

Therefore,

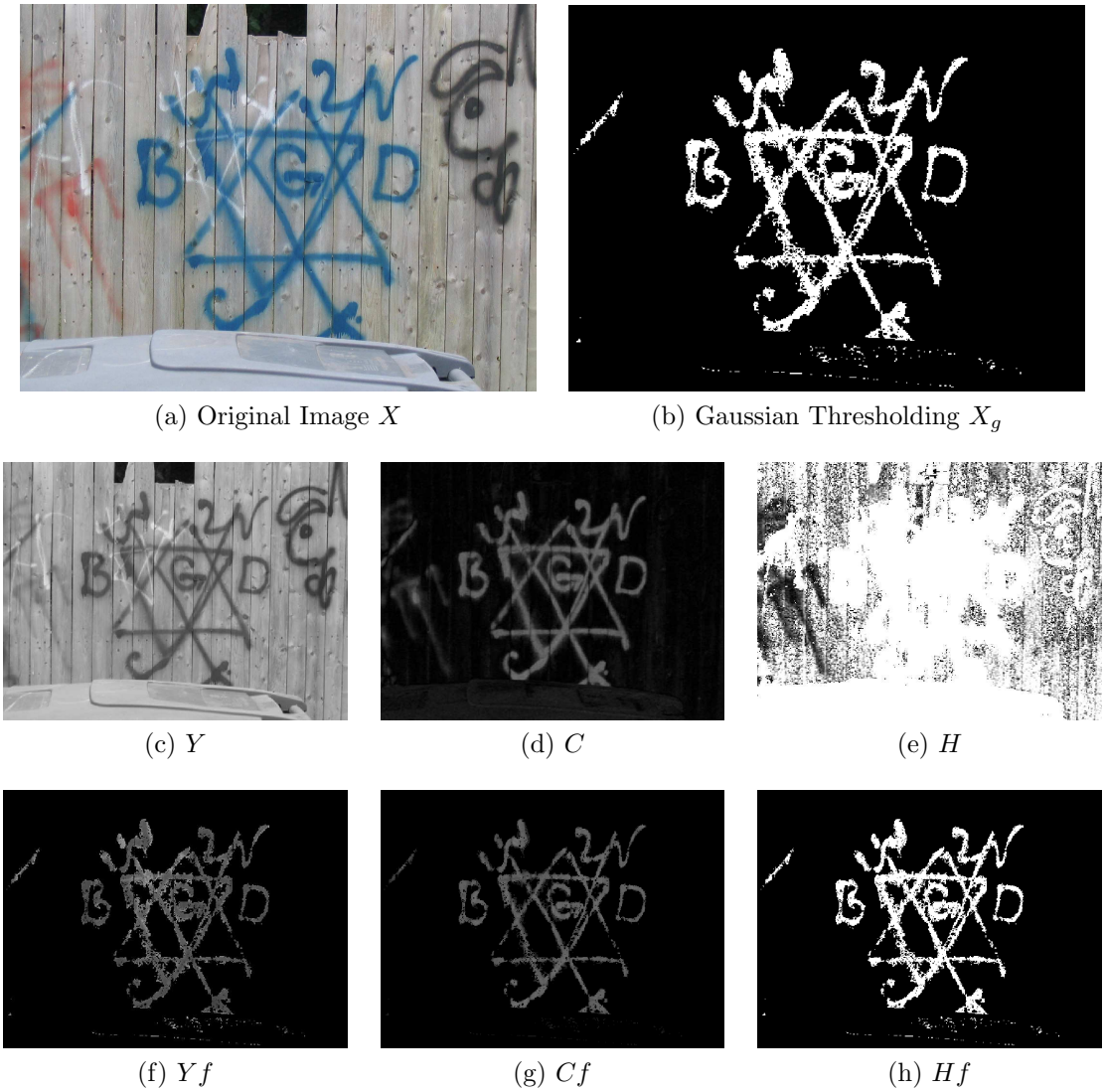
$$X_e = \begin{cases} X_e^Y \wedge X_e^C & \text{if } \frac{\sum_{b\in B}|\widetilde{Cf}_b^F - \widetilde{Cf}_b^B|}{n(B)} > T_e^C \\ X_e^Y \wedge X_e^H & \text{if } \widetilde{X} = \widetilde{H} \\ X_e^Y & \text{else} \end{cases}, \quad (3.25)$$

Table 3.6: Parameters and thresholds used in Block-Wise Gaussian Segmentation Enhancement. W_X and H_X are the width and height of X respectively.

Parameter	**Description**	**Value**
w_s	Block size for segmentation enhancement	$0.03max(W_X, H_X)$
T_e^C	Chroma threshold for channel enhancement	0.06

where $\wedge$ is the logical conjunction, also known as logical operator *and*. In the block diagram of Figure 3.15 the last module implements Equation 3.25 by doing $X_e = X_e^Y \wedge X_e^C \wedge X_e^H$, where X_e^C and X_e^H are set to an all-ones matrix $\mathtt{1}$ of the same size as X_e^Y if one or more of the conditions ($C1$, $C2$) are not satisfied. That is, if $C1$ is satisfied $X_e^H = \mathtt{1}$; if $C1$ is not satisfied and $C2$ is satisfied $X_e^C = \mathtt{1}$; if both $C1$ and $C2$ are not satisfied $X_e^C = X_e^H = \mathtt{1}$. Note that if we use the chroma channel enhancement we ignore the hue channel enhancement. This is because our experiments showed that if the condition for hue enhancement is satisfied the chroma enhancement does not improve the output. Also note how when $X_e = X_e^Y \wedge X_e^C$ the chrome enhancement can introduce some noise, which is removed using luma enhancement. Figures 3.16 and 3.17 show an example of the entire process. Note how X_e removes noise and enhances the graffiti, but also enhances some non-graffiti areas at the bottom. However, this areas will not be connected to graffiti components and we will be able to discard them in future steps.

Table 3.6 shows all the parameters/thresholds we used including empirically derived parameters.

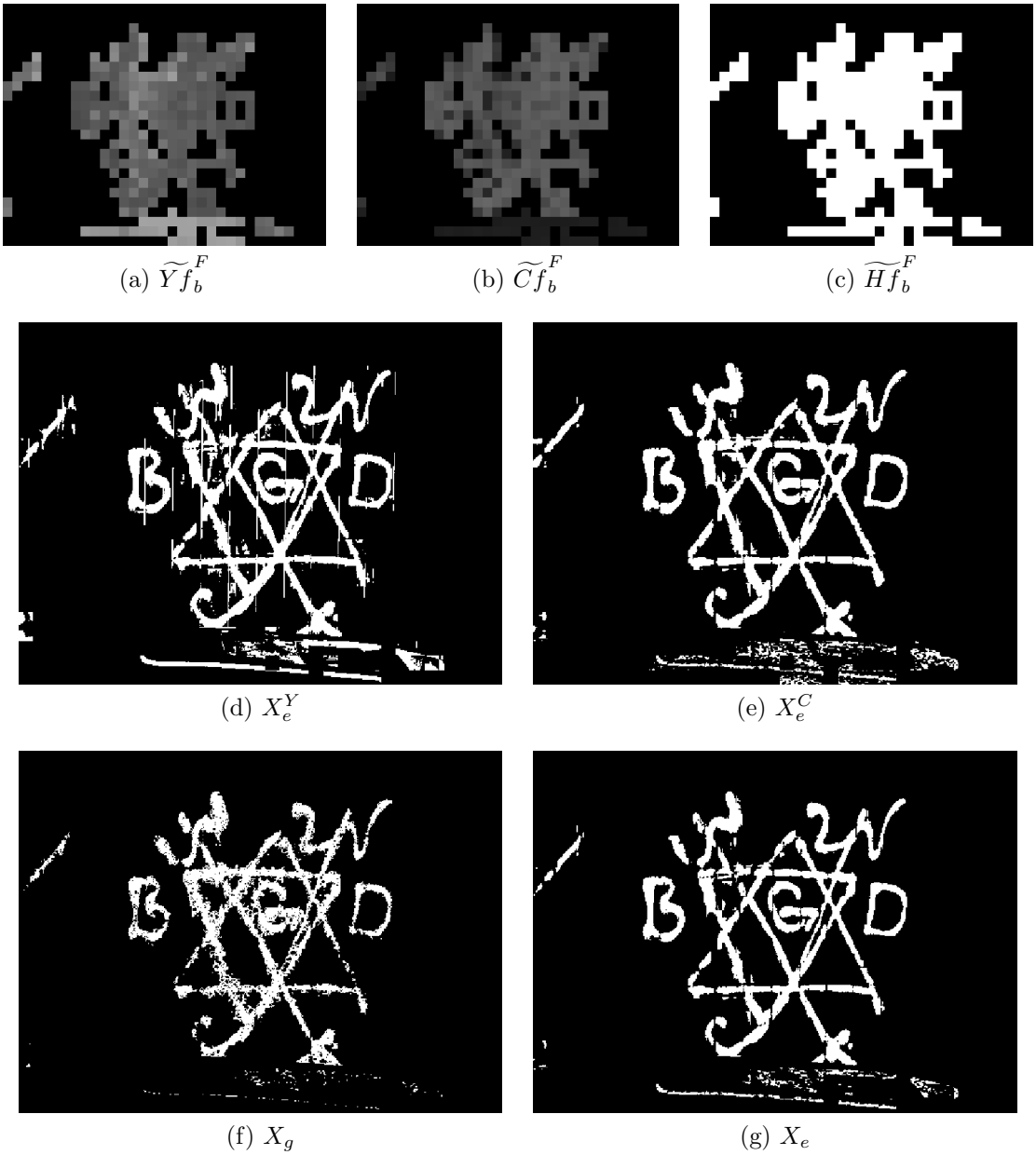


(a) Original Image X (b) Gaussian Thresholding X_g

(c) Y (d) C (e) H

(f) Yf (g) Cf (h) Hf

Fig. 3.16.: Example of Block-Wise Gaussian Segmentation Enhancement.



(a) $\widetilde{Yf}_b^F$ (b) $\widetilde{Cf}_b^F$ (c) $\widetilde{Hf}_b^F$

(d) X_e^Y (e) X_e^C

(f) X_g (g) X_e

Fig. 3.17.: Example of Block-Wise Gaussian Segmentation Enhancement (continued).

3.5.3 Background Stripe Removal

Gang graffiti are sprayed in all kinds of surfaces, including brick walls, garage doors and fences. All these surfaces contain stripes than can affect the graffiti component extraction. Figure 3.18 shows an example of a gang graffiti image after applying Block-Wise Gaussian Segmentation Enhancement. These stripes interfere with the segmentation by linking multiple gang graffiti components. Figure 3.19 shows the process to remove the background stripes. Note that sometimes the color of the background stripes is different from the graffiti itself, and the Color Image Segmentation Based on Gaussian Thresholding step already removes the stripes. Figure 3.20 shows an example.

(a) Original Image X

(b) Block-Wise Gaussian Segmentation Enhancement X_e

Fig. 3.18.: Background stripes affecting gang graffiti component segmentation.

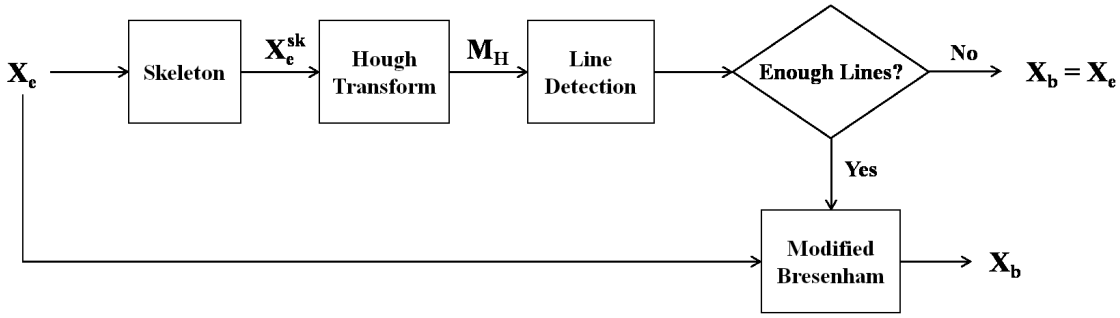


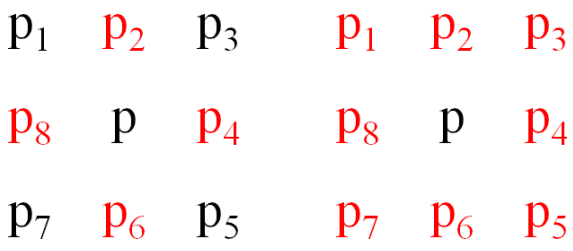
Fig. 3.19.: Background Stripe Removal.

(a) Original Image X

(b) Color Image Segmentation Based on Gaussian Thresholding X_g

Fig. 3.20.: Example of Background Stripes Removal During the Gaussian Thresholding Step.

First, we compute the skeleton X_e^{sk} of the input image X_e, the result of the Block-Wise Gaussian Segmentation Enhancement, which is binary. The skeleton is obtained using parallel thinning [224, 225] as follows. We define the set S as the set of all 1-valued pixels (ones) of X_e^{sk} representing objects (connected components) to be thinned. We define the set $\bar{S}$ as the set of all 0-values pixels (zeros) of X_e^{sk} representing either the background of or holes in S. The connectivities for S and $\bar{S}$ are set to 8-connectivity and 4-connectivity respectively. Figure 3.21 illustrates the meaning of 8-connectivity and 4-connectivity in a 3×3 support around a pixel p.



(a) 4-Connectivity (b) 8-Connectivity

Fig. 3.21.: Connectivity of p. Pixels are connected to p if they have the same value as p. Only pixel locations in red are considered in each connectivity.

We define $C(p)$ as the number of distinct 8-connected components of ones in p's 8-neighborhood. $C(p) = 1$ implies p is 8-simple when p is a boundary pixel [89]. We define $N(p)$ as

$$N(p) = min(N_1(p), N_2(p)), \tag{3.26}$$

where

$$N_1(p) = (p_1 \vee p_2) + (p_3 \vee p_4) + (p_5 \vee p_6) + (p_7 \vee p_8) \tag{3.27}$$

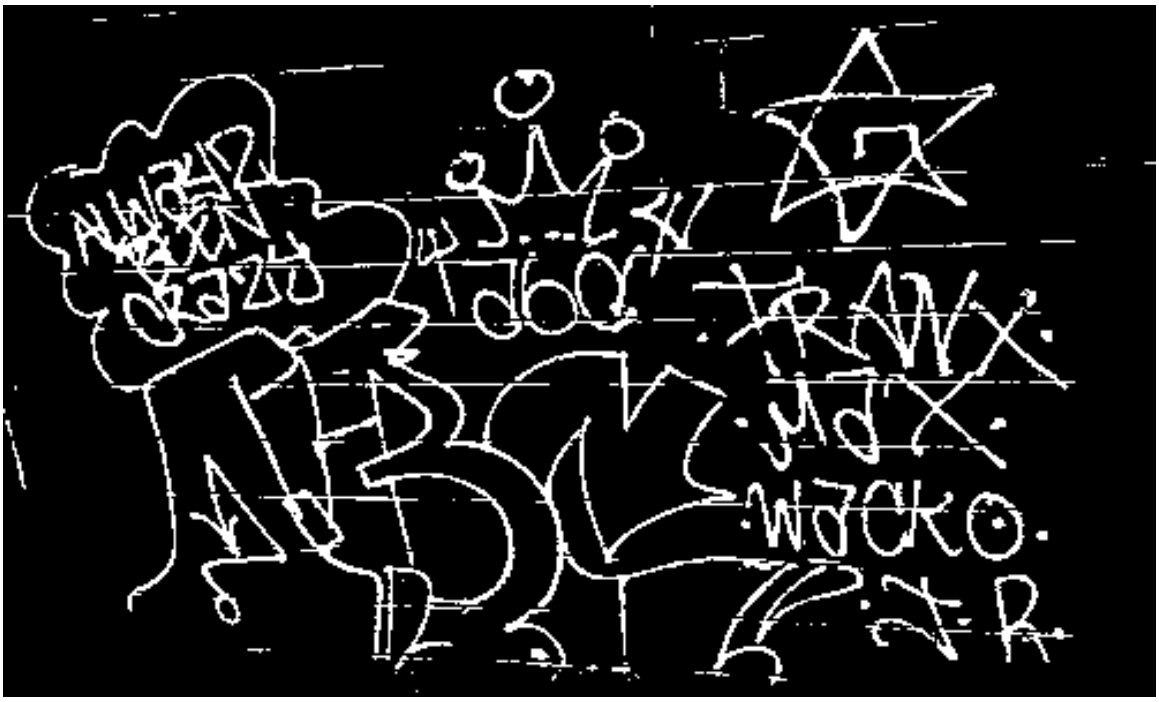
and

$$N_2(p) = (p_2 \vee p_3) + (p_4 \vee p_5) + (p_6 \vee p_7) + (p_8 \vee p_1). \tag{3.28}$$

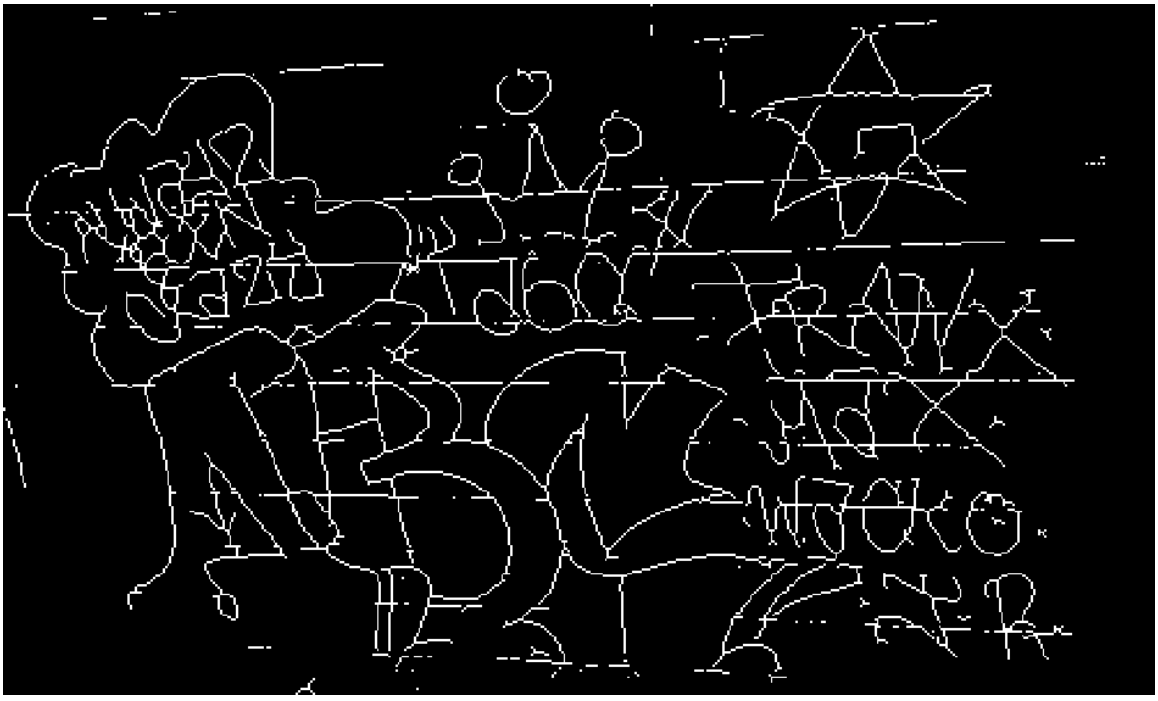
The symbols $\vee$ and $+$ are logical OR and arithmetic addition respectively. Note that $N_1(p)$ and $N_2(p)$ divide the ordered set of neighbors of p into four pairs of adjoining pixels and count the number of pairs that contain one or two ones. The thinning process is applied to each pixel $p \in S$. p is deleted (i.e. changing one to zero) if all the following conditions are met:

1. $C(p) = 1$
2. $T^L_{N(p)} \leq N(p) \leq T^H_{N(p)}$
3. Either
   (a) $(p_2 \vee p_3 \vee \bar{p_5}) \vee p_4 = 0$ in odd iterations
   (b) $(p_6 \vee p_7 \vee \bar{p_1}) \wedge p_8 = 0$ in even iterations

where $T^L_{N(p)} = 2$, $T^H_{N(p)} = 3$, and $\bar{p}$ and $\wedge$ are logical complement and logical AND respectively. The thinning stops when no further deletions are possible. Figure 3.22 shows an example of skeletonization via parallel thinning to obtain X^{sk}_e.



(a) Binary Image X_e



(b) Parallel Thinning X_e^{sk}

Fig. 3.22.: Skeletonization via Parallel Thinning [225].

The next step is to find straight lines using the Standard Hough Transform (SHT) [226, 227]. The method uses the parametric representations of a line to populate a 2-dimensional matrix M_H called accumulator array, where its rows and columns correspond to ρ and θ values of $\rho = x\cos(\theta) + y\sin(\theta)$ respectively. Figure 3.23 illustrates the parametric representation of a line.

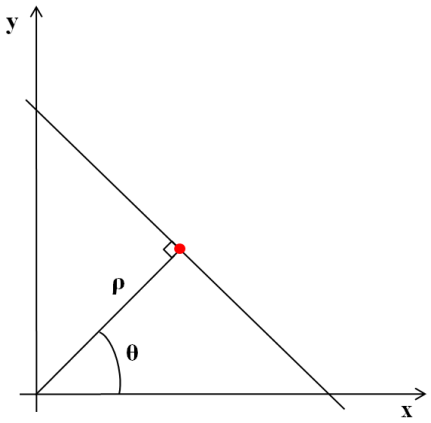
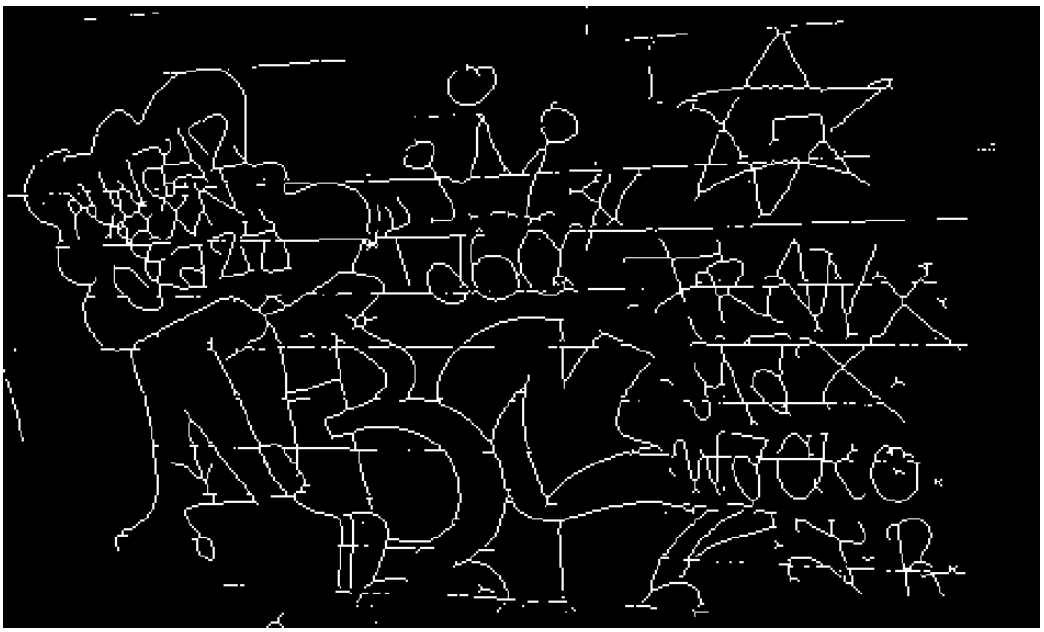


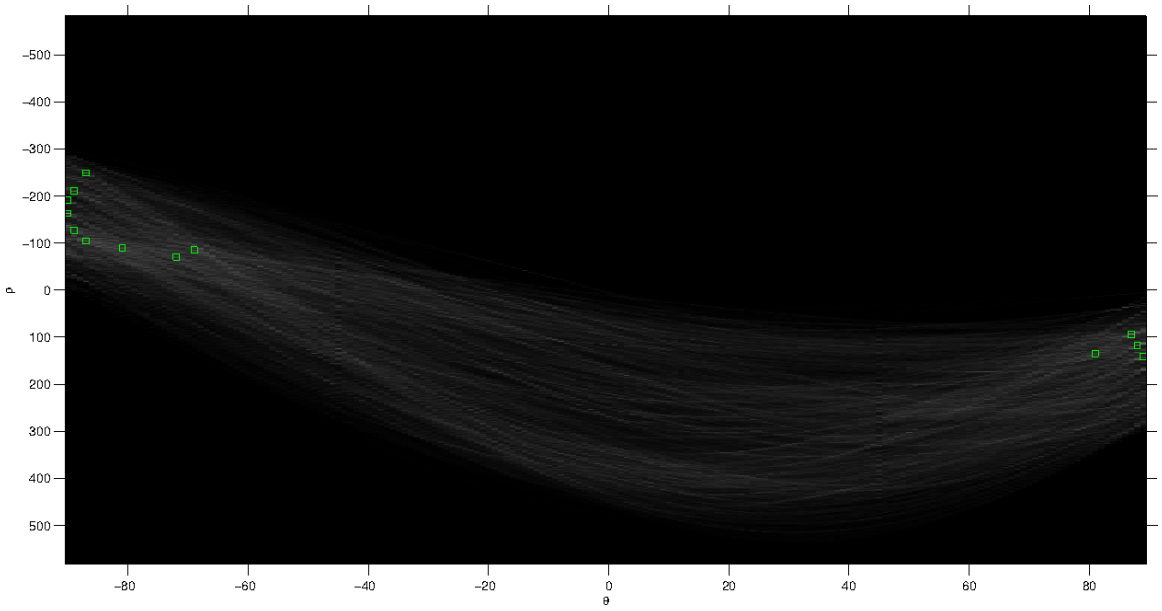
Fig. 3.23.: Parametric Representation of a Line.

First, each cell in M_H is initialized to zero. For each non-zero pixel in X_e^{sk} the accumulator cells are updated so that $M_H(i, j)$ keeps a count of the number of pixels in the XY plane represented by $\rho(i)$ and $\theta(j)$. Peak values in M_H represent potential lines in X_e^{sk}. We Figure 3.24 shows the Hough accumulator array M_H with highlighted peaks. There are 13 potential lines divided in two sets of θ around π and $-\pi$, which actually correspond to the same set.

Given the nature of the background stripes in gang graffiti images we limit the number of peaks to $N_{peaks} = 15$. For each peak we find the location of all nonzero pixels in the image that contributed to that peak and determine the line segments based on those pixels. Each segment is now represented by a set (θ, ρ, p_i, p_f), where (p_i, p_f) are the initial and final points of the segment. We discard segments of length less than $T^W_{minlen} = 0.4W_X$ if the segment is closer to the horizontal plane and less than $T^H_{minlen} = 0.6H_X$ if the segment is closer to the vertical plane. W_X and H_X are the width and the height of the image, respectively. If we have less than $N_{seg} = 4$



(a) Skeleton X_e^{sk}



(b) M_H

Fig. 3.24.: Standard Hough Transform accumulator array. Peaks corresponding to potential lines are marked with green squares.

segments remaining we consider them not to be background stripes, and there is nothing to be done. Else, we need to remove the segments without affecting the graffiti components they may intersect with.

To do that we propose a modification of the Bresenham's technique [228]. The original method retrieves a set of pixels locations $S_{(x,y)}$ from a given line represented by a set of initial and final points (p_i, p_f). Figure 3.25 illustrates the conversion from (p_i, p_f) to $S_{(x,y)}$. The pixels in $S_{(x,y)}$ are marked in gray. Figure 3.26 shows a step of the process when a pixel location (shown in yellow at (x, y)) has been already added to $S_{(x,y)}$. Since the line does not fall into the actual pixel grid the next sampled location (shown in yellow at $(x + 1, y + 1)$) will have an error ϵ on the y direction. Note that this error ranges between -0.5 to 0.5. The next point to be added to $S_{(x,y)}$ can either be $(x + 1, y)$ or $(x + 1, y + 1)$. We choose $(x + 1, y)$ if $y + \epsilon + m < y + 0.5$, and we choose $(x + 1, y)$ otherwise. By doing so we minimize the total error between the mathematical line segment and what we actually add to $S_{(x,y)}$.

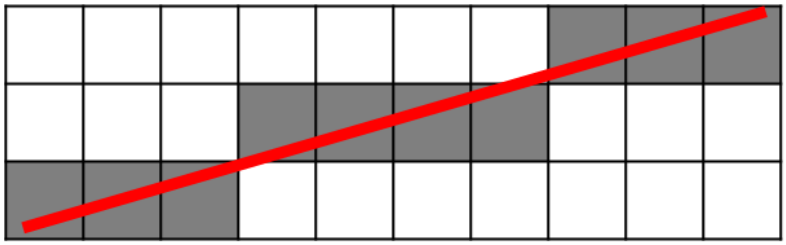


Fig. 3.25.: Bresenham's Technique: mathematical line (red) and elements of $S_{(x,y)}$ (gray).

Our modification to the original method includes an estimation of the segment width at each new location added to $S_{(x,y)}$. For this purpose we need to use the binary image X_e instead of its skeleton X_e^{sk}. At each new location (x, y) we create a window of radius 1 around it and compute the ratio R as

$$R = \frac{\#\text{ ones inside window}}{\#\text{ pixels inside window}} \tag{3.29}$$

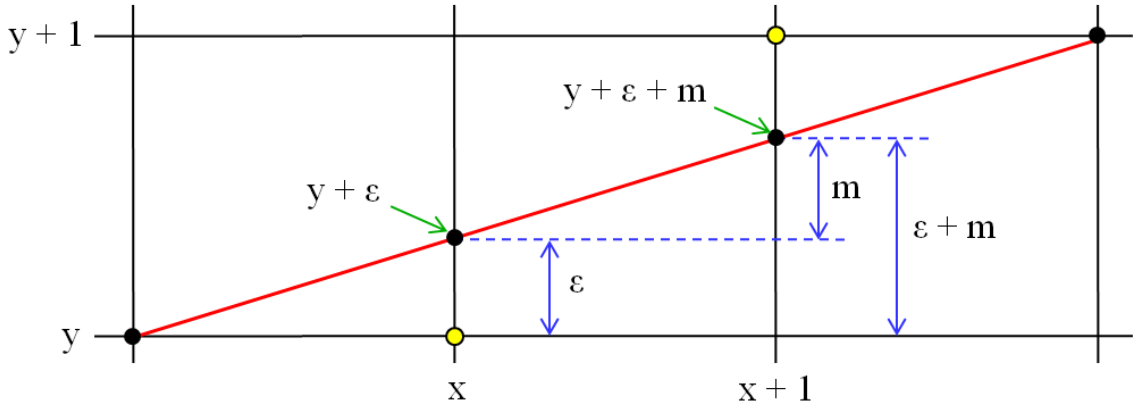


Fig. 3.26.: Step of Bresenham's Technique.

If $R > T^H_{rad}$ we increase the window size by one and recompute R. We repeat the process until $R \leq T^H_{rad}$. We choose $T^H_{rad} = 0.6$ as it gave us the best results in our experiments. Figure 3.27 illustrates the final size of the window at different locations. Note that even though the segment can have an arbitrary orientation the window is always aligned with the XY axes. This is because we just need an estimate of the segment width.

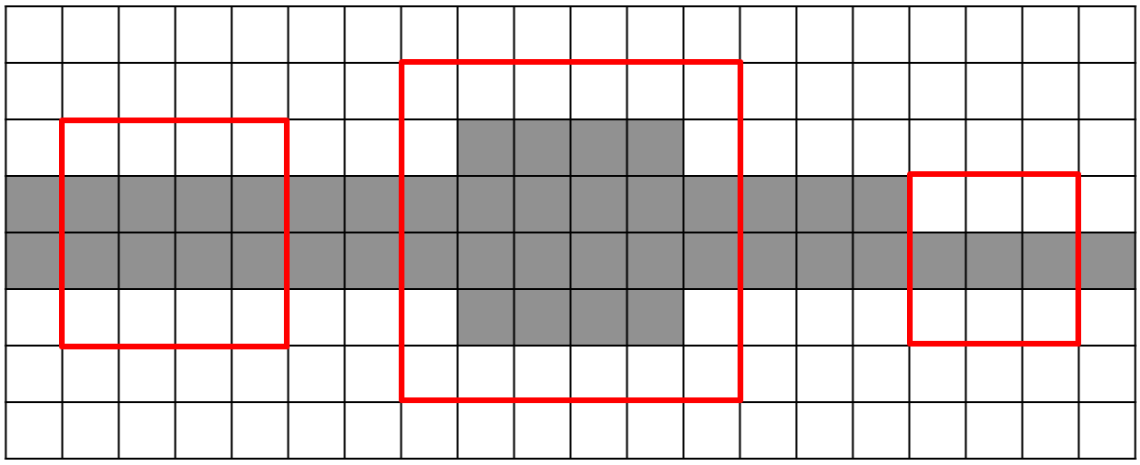


Fig. 3.27.: Final window sizes at different locations using our modified Bresenham's Technique.

Once we have all the segment width estimates for all the pixel locations in $S_{(x,y)}$ we set the segment width to the most frequent estimated width (i.e. the width mode). The pixel locations with width larger than the mode are considered to be intersections with graffiti components, and they are left untouched. The rest of the pixel locations are removed from the binary image. After all the line segments are processed we obtain the binary image X_b. Figure 3.28 shows an example of our proposed modified Bresenham's Technique. The green areas correspond to removed line segments, and the blue areas correspond to ignored line segments. Figure 3.29 shows an example of the entire Background Stripe Removal process. Note how some of the line segments actually corresponding to background stripes are not removed. However, we have removed the segments that connect different graffiti components, and they can now

Table 3.7: Parameters and thresholds used in Background Stripe Removal. W_X and H_X are the width and height of X respectively.

Parameter	Description	Value
$T^L_{N(p)}$	Low threshold for thinning	2
$T^H_{N(p)}$	High threshold for thinning	3
N_{peaks}	Number of Hough peaks	15
T^W_{minlen}	Threshold to discard horizontal segments	$0.4W_X$
T^H_{minlen}	Threshold to discard vertical segments	$0.6H_X$
N_{seg}	Number of segments to keep	4
T^H_{rad}	High threshold for line width	0.6

be separated.

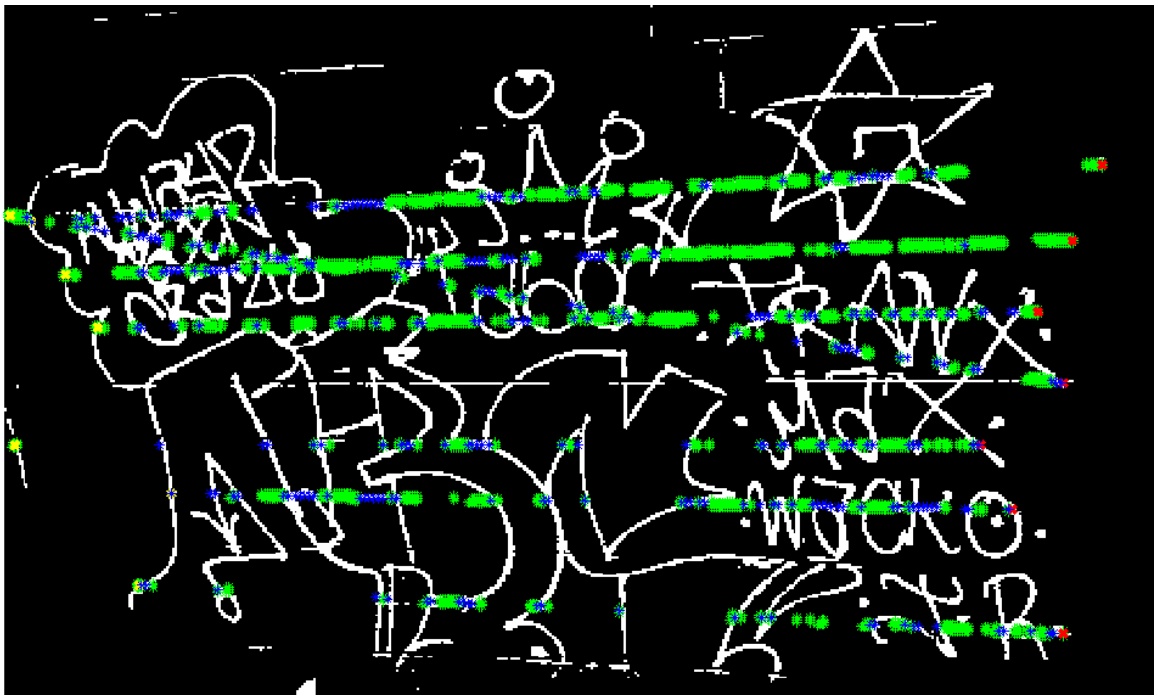
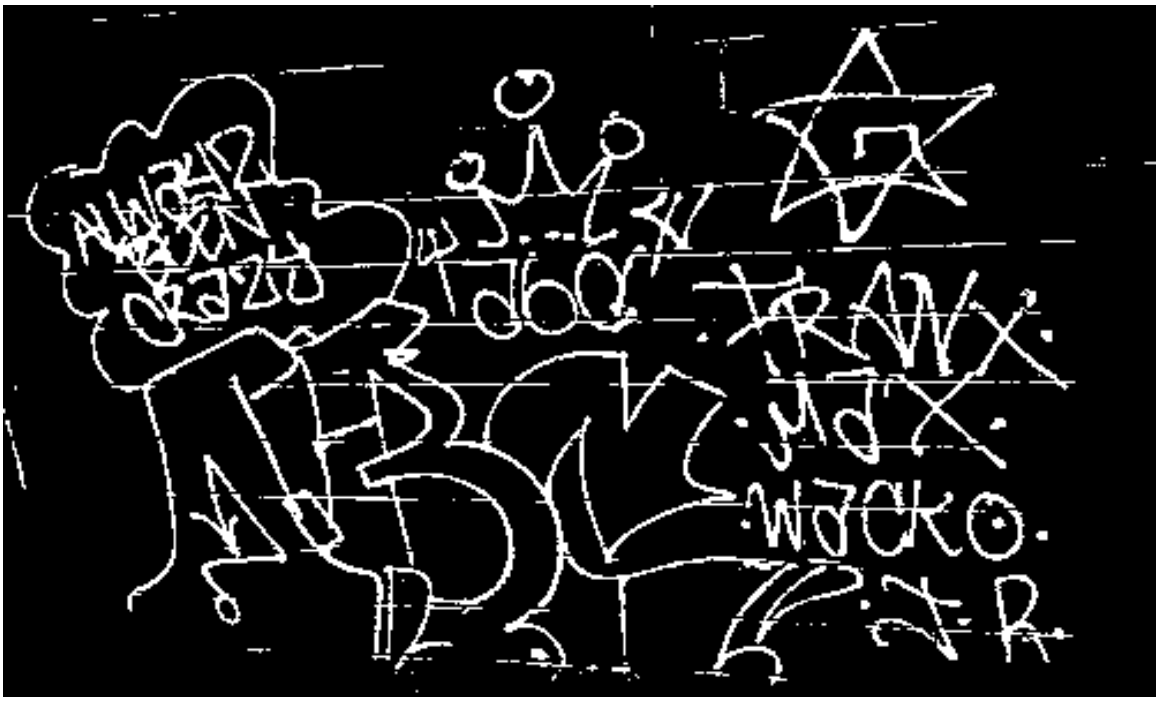
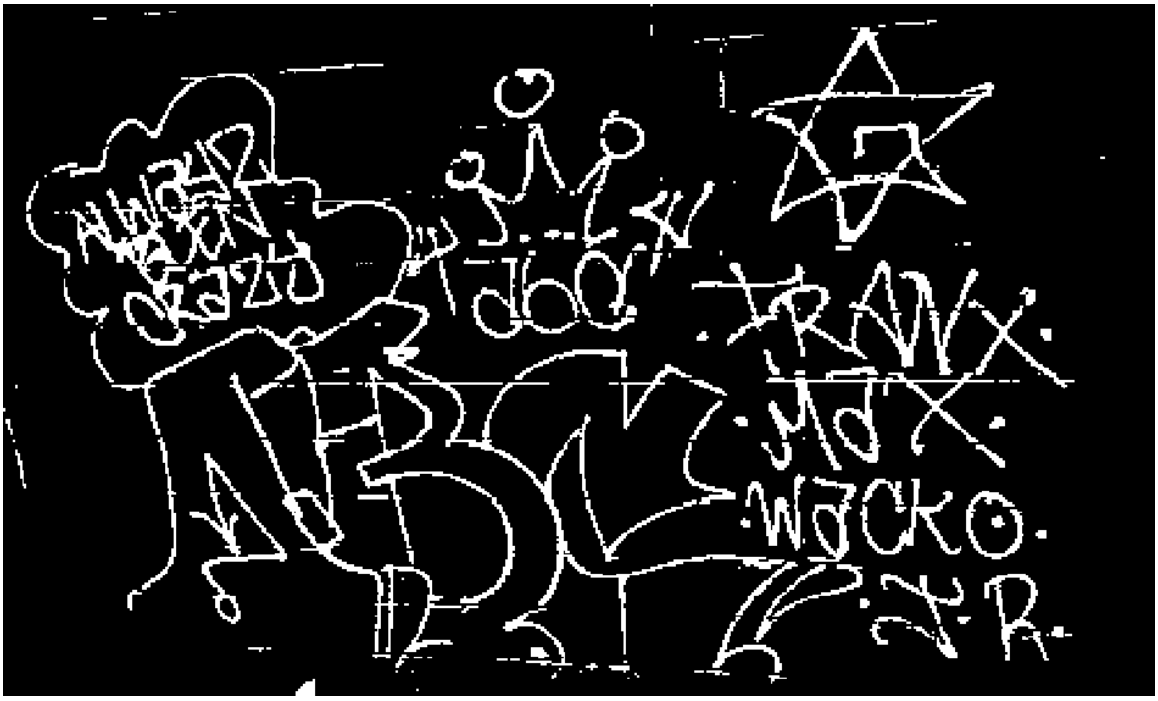


Fig. 3.28.: Modified Bresenham Technique. Green areas correspond to removed line segments; blue areas correspond to ignored line segments.

Table 3.7 shows all the parameters/thresholds we used including empirically derived parameters.



(a) Input: X_e



(b) Output: X_b

Fig. 3.29.: Example of Background Stripe Removal.

3.5.4 Graffiti Component Reconnection

Even after Block-Wise Gaussian Segmentation Enhancement and Background Stripe Removal there are still broken gang graffiti components that need to be reconnected for efficient segmentation. For this purpose we consider a line reconstruction method used in topographic map enhancement [147, 229]. Figure 3.30 shows the process to reconnect graffiti components.

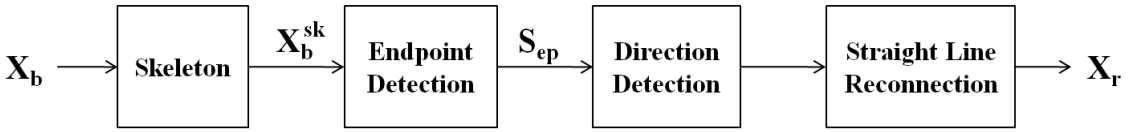


Fig. 3.30.: Graffiti Component Reconnection.

First, we compute the skeleton X_b^{sk} of the input image X_b, the result of the Background Stripe Removal, which is already binary. The skeleton is obtained using the method already described in Section 3.5.3. We then detect the endpoints of X_b^{sk}. An endpoint is defined to have exactly one neighbor pixel. Figure 3.31 illustrates all the possible 3×3 templates of an endpoint. Figure 3.32 shows an example of detected endpoints.

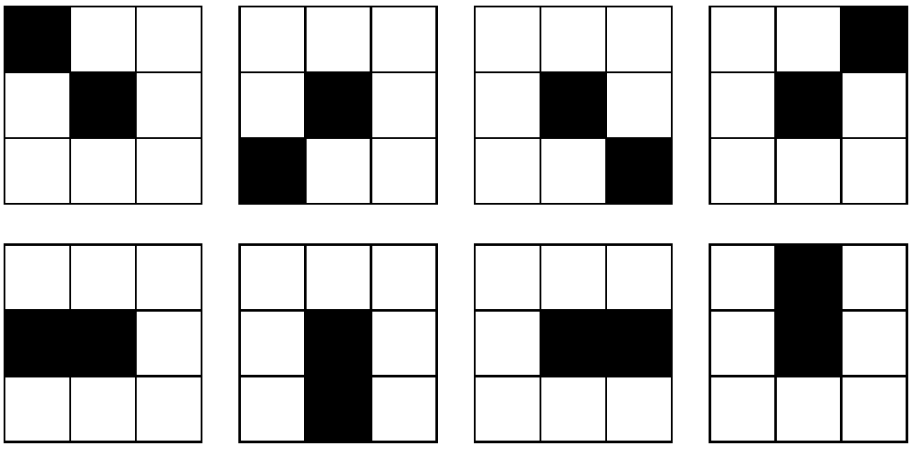


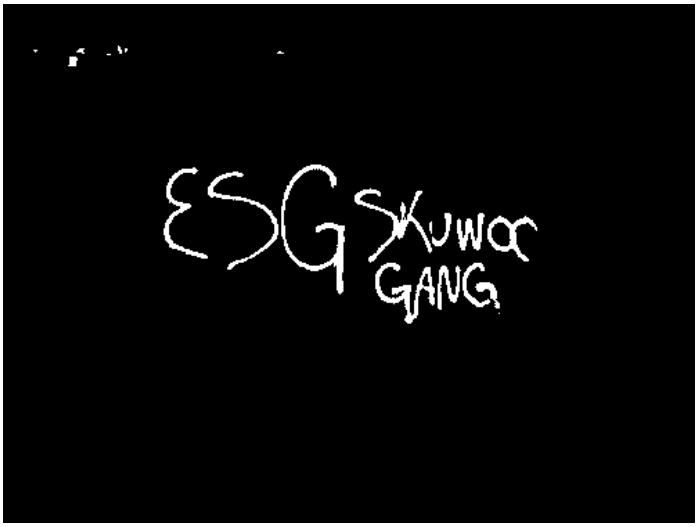
Fig. 3.31.: 3×3 templates to detect an endpoint. The endpoint is at the center of the template.

For each endpoint e_0 we create a $h \times h$ search window around it and build the set $S_{ep} = \{e_1, e_2, ..., e_n\}$ with the n endpoints within the search window. We selected $h = 20$ as proposed in [147]. Note that we ignore any endpoints that are 8-neighbor connected to e_0 (i.e. part of the same connected component). For each endpoint $e_i \in S_{ep}$ we detect its direction by constructing a chain code as shown in Figure 3.33. We backtrace $N_{px}^{bt} = 5$ pixels and assign a zone based on the possible directions $0 - 7$ according to Table 3.8.

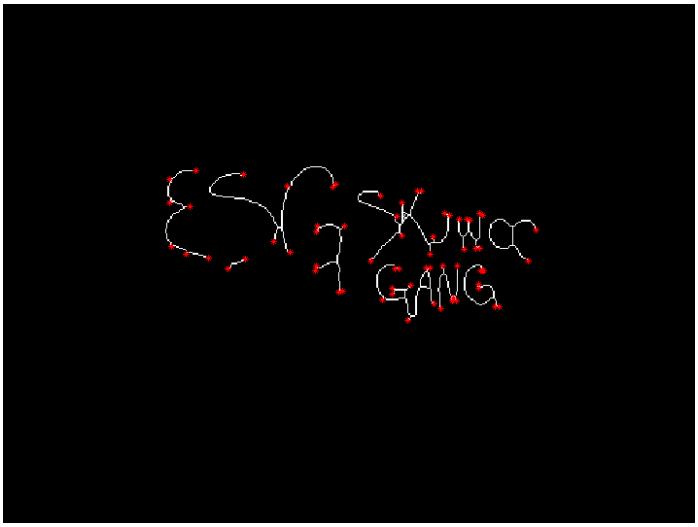
Table 3.8: Relationship Between Directions and Zones in the Chain Code.

Directions	**Zone**
1, 2	Zone 1
3, 4	Zone 2
5, 7	Zone 3
7, 8	Zone 4

(a) Original Image X



(b) X_b



(c) Endpoints on X_b^{sk}

Fig. 3.32.: Endpoint Detection.

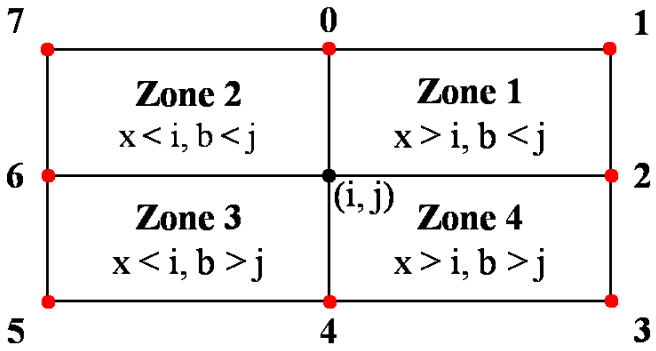


Fig. 3.33.: Chain Code For Endpoint Direction Detection.

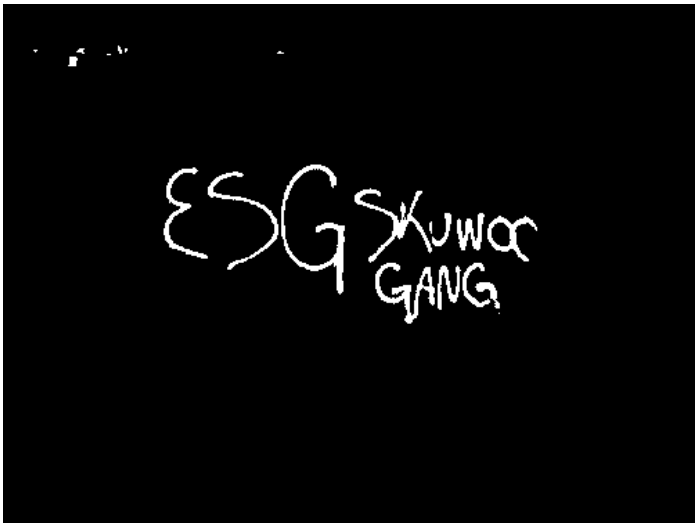
We remove from S_{ep} all the endpoints that do not satisfy the following conditions with respect to e_0:

- For opposite directions:
  - Zone 1 opposite to Zone 3
  - Zone 2 opposite to Zone 4
- For parabolic directions:
  - Zone 1 parabolic with Zone 4
  - Zone 4 parabolic with Zone 3
  - Zone 3 parabolic with Zone 2
  - Zone 2 parabolic with Zone 1

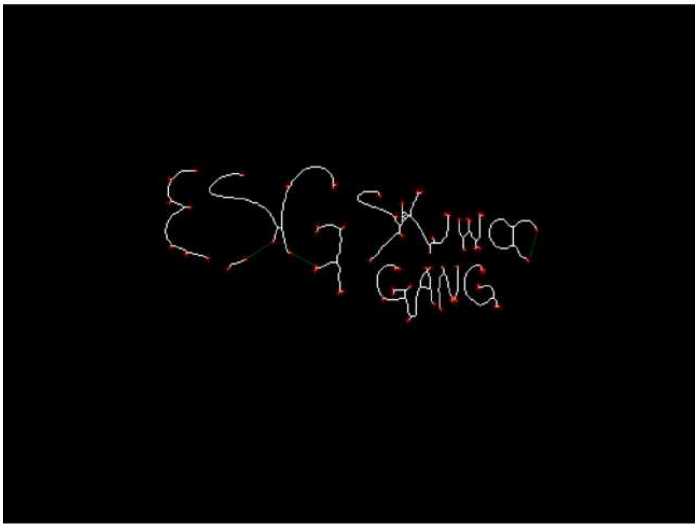
If there are more than one remaining endpoints in S_{ep} we chose the one closest to e_0, e_d. The method presented in [147] does reconnection between e_0 and e_d with Cubic Spline Interpolation or Newton Interpolation Method [230]. Since we are just interested in combining disconnected components for classification and not reconstructing them we reconnect e_0 and e_d with a straight line. After all the endpoints are processed we obtain the binary image X_r. Figure 3.34 shows an example of the Graffiti Component Reconnection process.

At this point each individual graffiti component corresponds to an 8-neighbor connected component. Figure 3.35 shows an example of the connected component extraction before and after the Automatic Graffiti Component Segmentation. Note that currently we do not try to connect different letters on the same word. Given the handwritten nature of the graffiti (e.g. "y" in Figure 3.35b) it is difficult to discern between words and symbols.

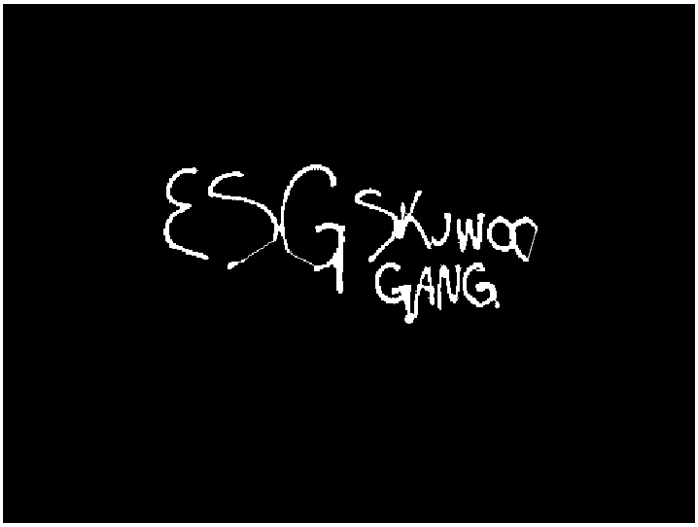
Note that this method can also be used to reconstruct graffiti components that are broken because of being crossed-out by other graffiti component sprayed using



(a) X_b



(b) Reconnected Components.



(c) X_r

Fig. 3.34.: Example of Graffiti Component Reconnection.

(a) Connected Components of X_g (Output of Gaussian Thresholding)

(b) Connected Components of X_r (Output of Graffiti Component Reconnection)

Fig. 3.35.: Example of connected components after Gaussian Thresholding and after Graffiti Component Reconnection.

Table 3.9: Parameters and thresholds used in Graffiti Component Reconnection.

Parameter	**Description**	**Value**
h	Endpoint search window size	20
N_{px}^{bt}	Number of backtracing pixels	5

different color.

Table 3.9 shows all the parameters/thresholds we used including empirically derived parameters.

3.6 Gang Graffiti Features

The GARI system provides gang graffiti image retrieval in two scenarios: 1) recognize scenes containing graffiti and 2) classify individual graffiti components. We explain both scenarios in detail in Section 3.7.

For scene recognition we find SIFT features from the entire image, similar to the work done in [7, 8, 10–12, 18] for graffiti and tattoo images. SIFT is invariant to location, scale and rotation, and it is robust to affine transformations and illumination changes and viewpoint. The process to create SIFT descriptors from an image can be summarized as follows.

First, we find all the local extrema in the Difference of Gaussian (DoG) pyramid [15, 231]. A Gaussian pyramid for an image is generated by smoothing it with successively larger Gaussian functions

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \tag{3.30}$$

and arranging the sequence of smoothed images in the form of a stack. Each level of the Gaussian pyramid is one octave above the level below (i.e. doubling the value of σ). A DoG image $D(x, y, \sigma)$ at scale σ is defined as

$$D(x, y, \sigma) = L(x, y, k_i\sigma) - L(x, y, k_j\sigma), \tag{3.31}$$

where $L(x, y, k\sigma)$ is the convolution of the original image with $G(x, y, k\sigma)$. Figure 3.36 illustrates how the DoG pyramid is generated.

The local extrema (keypoints) are detected from the subpixel minima/maxima in the DoG pyramid by comparing neighboring pixels across scales, as shown in Figure

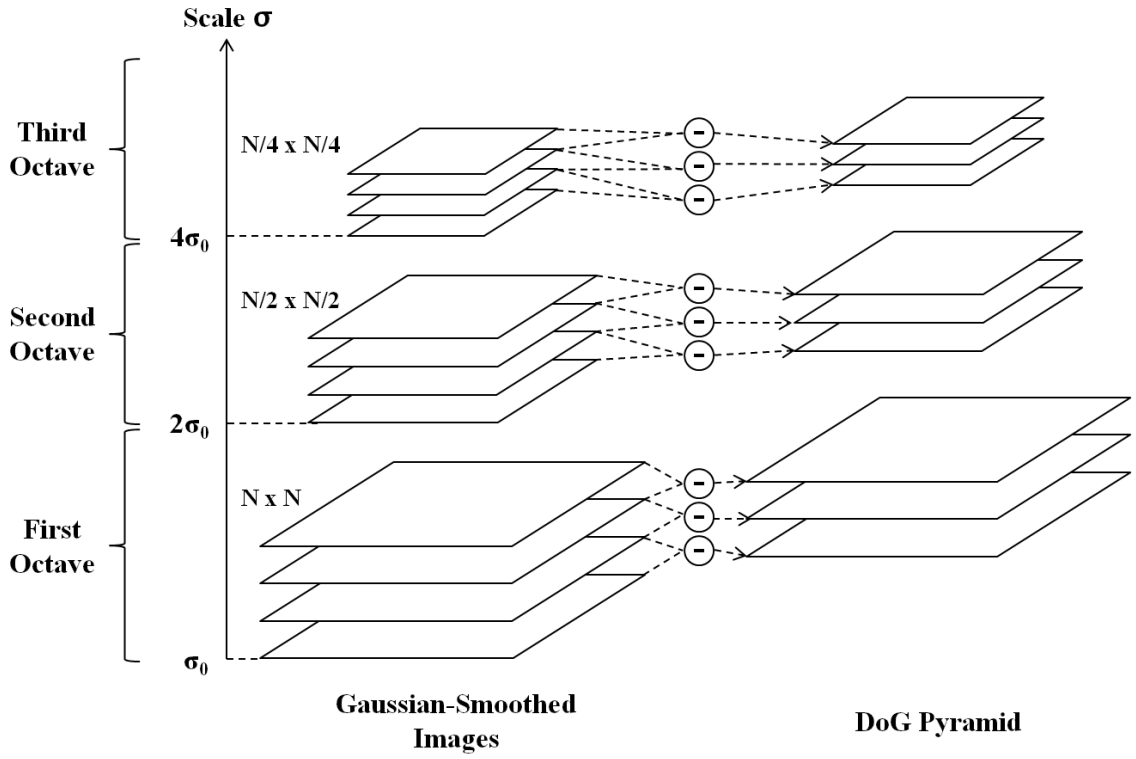


Fig. 3.36.: DoG Pyramid.

3.37. The subpixel accuracy is interpolated using the quadratic Taylor expansion of $D(x, y, \sigma)$ with the candidate keypoint $x = (x, y, \sigma)$ as the origin:

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D^T}{\partial x^2} x \tag{3.32}$$

Weak extrema are discarded by rejecting keypoints that satisfy $|D(x)| < 0.03$.

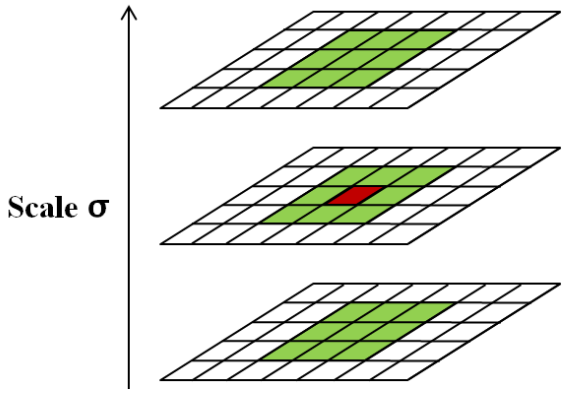
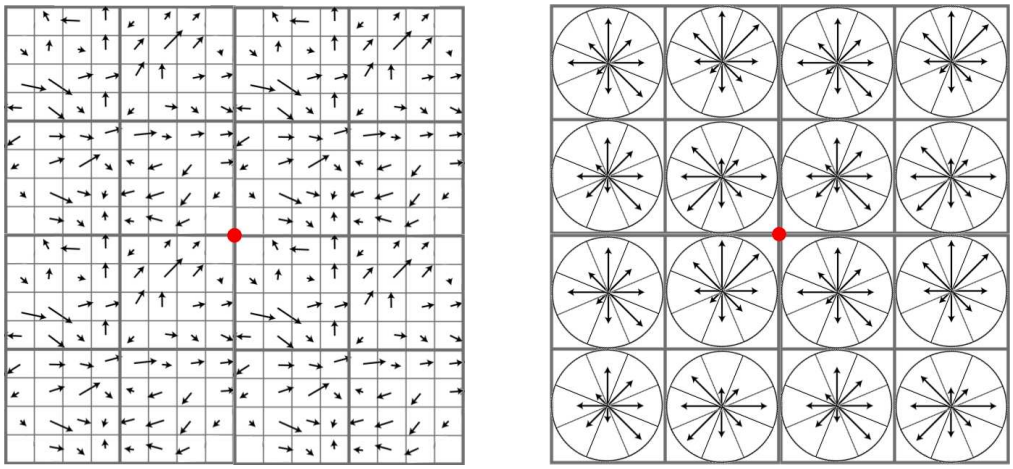


Fig. 3.37.: Neighboring Pixels (green) For Keypoint Extraction (red).

We then associate a dominant local orientation to a keypoint by constructing a histogram of gradient orientations using 36 bins spanning 360°. The bin with where the histogram peak occurs decides the dominant local orientation. By representing the keypoint relative to its orientation the SIFT descriptor achieves rotation invariance.

Finally, the SIFT descriptor is created by surrounding each keypoint with a 16×16 descriptor window divided into 4×4 cells. The gradient magnitudes in the descriptor window are weighted by a Gaussian function with σ equal to half the width of the neighborhood. For each of the 16 cells an 8-bin orientation histogram is determined, thus creating a 128-dimensional descriptor with its length normalized to make it robust to changes in illumination. Figure 3.38 shows a graphical representation of the keypoint descriptor generation. Figure 3.39 shows some examples of extracted SIFT keypoints overlapped on the input images.



(a) Gradient Magnitudes Around Descriptor Window

(b) 16 8-Bin Orientation Histograms (128-dimensional)

Fig. 3.38.: Keypoint Descriptor Generation. The red dot represents the location of the keypoint.

(a) SIFT Descriptors

(b) Gradient Magnitude Histograms

Fig. 3.39.: 25 SIFT descriptors selected at random. Each keypoint is represented by a set of gradient magnitude histograms (green) rotated to its dominant local orientation (yellow). The size of the green grid represents the scale of the descriptor.

For individual gang graffiti component classification we do not use SIFT descriptors directly, but the spatial locations of the SIFT keypoints to create Local Shape Context (LSC) descriptors similar to the work proposed in [30, 183]. We do this because graffiti components are handwritten shapes with intra-class inconsistencies and small shape distortions that are not fully captured with SIFT descriptors. Also, SIFT descriptors accommodate for illumination changes and complex textures, which are not present in binarized graffiti components.

First, we find the gang graffiti components as individual connected components from the output of the Automatic Graffiti Component Segmentation in Section 3.5. For each graffiti component we then find N_f SIFT keypoint locations. Each location f_i needs to be compared against the other $N_f - 1$ locations to create a LSC descriptor. This is done by binning the locations into a histogram, where its bins are broad enough to allow for small shape distortions and orientation variation. Our proposed histogram is defined with $n_r = 3$ concentric circumferences representing log-radial distance bins and $n_\theta = 16$ equally spaced sectors representing angles. We use a log-radial increment because we want to give more importance to the neighbor features than the rest. A histogram is centered at f_i and its bins are populated by calculating the distances

$$r_i^j = \frac{||f_i - f_j||_2}{\bar{r_i}} \tag{3.33}$$

and the angles

$$\bar{\theta}_i^j = \theta_i^j - \theta_i, \tag{3.34}$$

Table 3.10: Parameters and thresholds used for the Gang Graffiti Features.

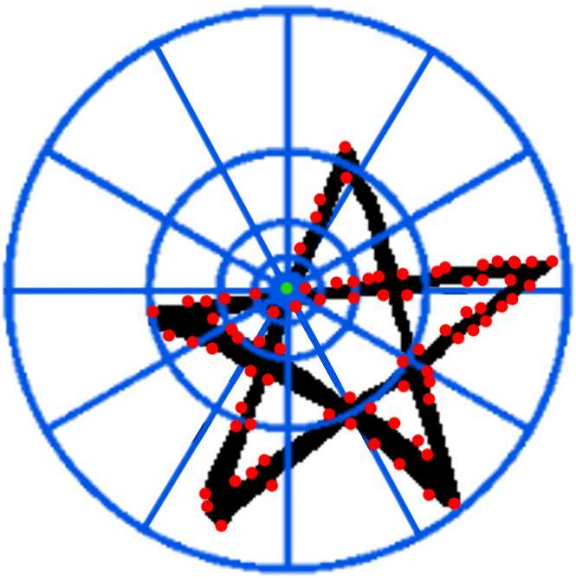
Parameter	**Description**	**Value**
n_r	Number of log-radial distance bins	6
n_θ	Number of angular bins	19

for all $j \in [1, N_f]$ and $j \neq i$, where $\bar{r_i}$ is the average distance between f_i and the rest of locations, θ_i^j is the angle between f_i and f_j, and θ_i is the dominant local orientation already described. Note that θ_i^j can be determined by

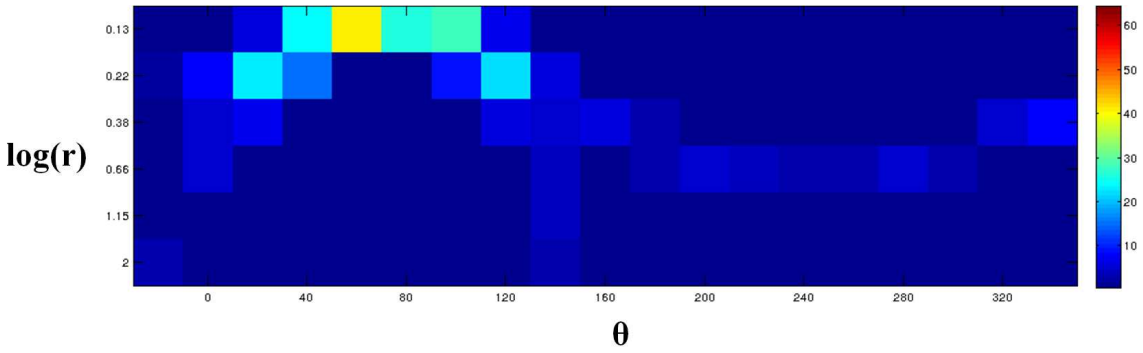
$$\arctan \frac{f_{iy} - f_{jy}}{f_{ix} - f_{jx}}, \tag{3.35}$$

where f_{kx} and f_{ky} are the x and y components of the k^{th} location. By normalizing r_i^j by $\bar{r_i}$ and subtracting θ_i from $\bar{\theta}_i^j$ we achieve scale invariance and rotation invariance in the LSC descriptor respectively. Each LSC histogram is then represented a normalized $n_r \times n_\theta$ matrix, which can be flattened to a $n_r n_\theta$-dimensional descriptor. Figure 3.40 illustrates the histogram and the distributions of the bins overlaid on a gang graffiti component.

Table 3.10 shows all the parameters/thresholds we used including empirically derived parameters.



(a) LSC Log-Radial Histogram



(b) LSC Normalized Matrix

Fig. 3.40.: Local Shape Descriptor histogram for a specific keypoint and its matrix representation. The matrix holds the count distribution of SIFT keypoint locations relative the specific keypoint.

3.7 Content Based Gang Graffiti Image Retrieval

We describe a method to recognize gang graffiti by matching image features from query images against our database of gang graffiti. The method is currently used in two scenarios: 1) "Gang Graffiti Scene Recognition" to recognize scenes containing graffiti (Figure 3.41) and 2) "Gang Graffiti Component Classification" to classify individual graffiti components (Figure 3.42). In both cases we use a vocabulary tree [196] to retrieve input images.

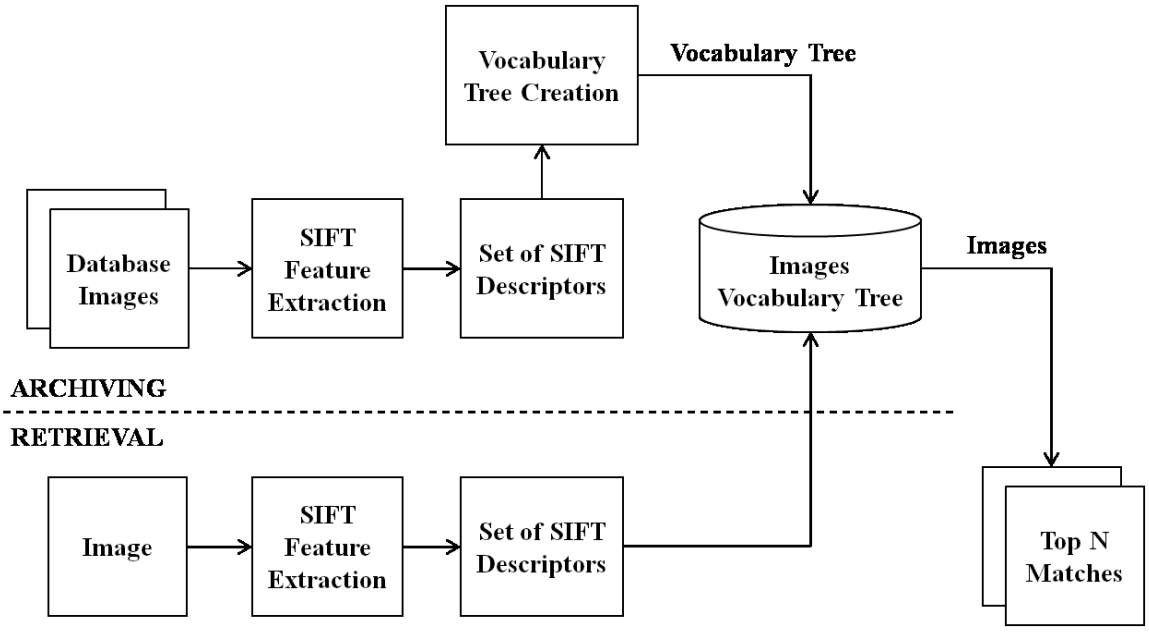


Fig. 3.41.: Gang Graffiti Scene Recognition.

The vocabulary tree is obtained as follows. First, we find features from a set of database images to get N D-dimensional vectors (i.e. descriptors), where D will depend on the type of feature [15, 24, 232, 233]. All the N D-dimensional descriptors populate the R^D space, which we then recursively divide into sub-clusters using hierarchical k-means clustering [234].

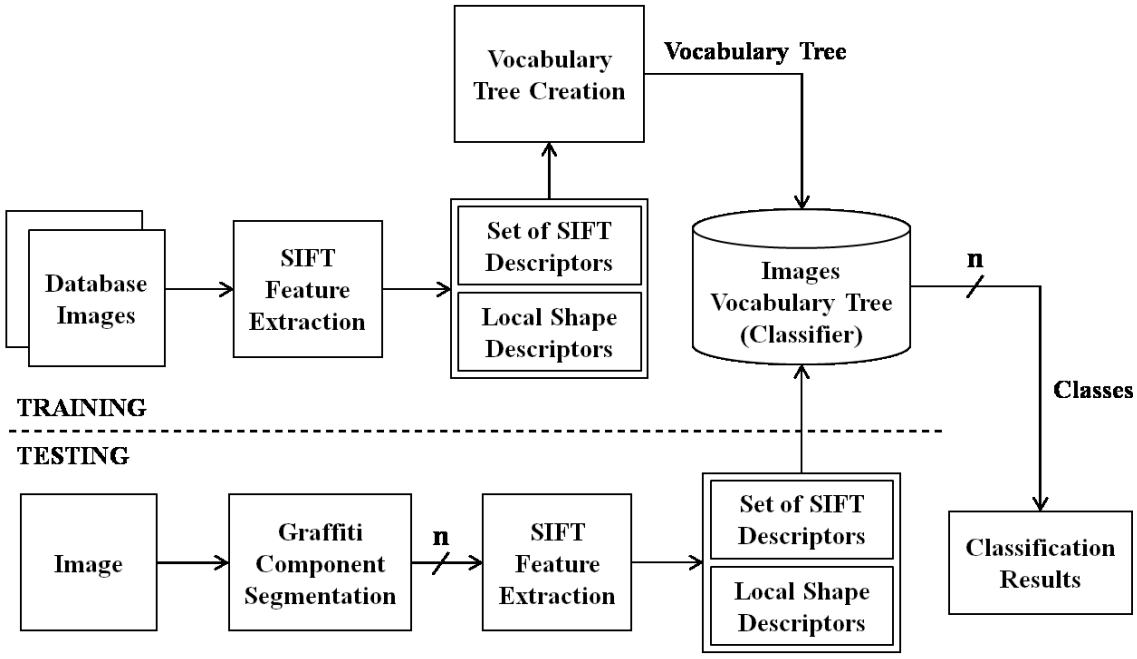


Fig. 3.42.: Gang Graffiti Component Classification.

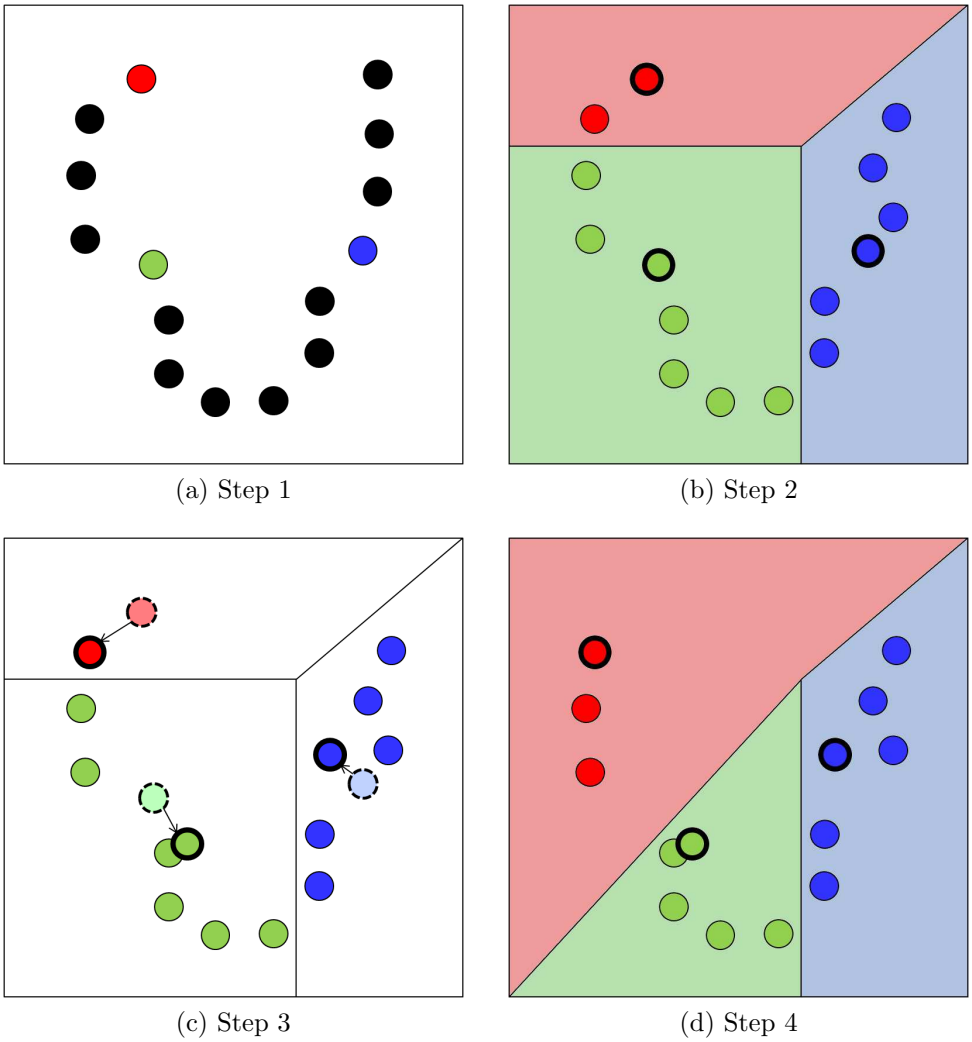
At each recursion level k-means is used in four steps. First, k initial "means" are randomly chosen among all the data in the cluster. Second, k clusters are created by associating every data sample to its nearest mean. Third, each cluster is given a new mean computed as the centroid of all the data points associated with it. Finally, the second and third steps are repeated until convergence is reached (no data sample moves from one cluster to another). Figure 3.43 illustrates the entire process. Since k-means is greedy for minimizing the sum of squared errors (SSE) it may not converge to the global optimum. Its performance strongly depends on the initial guess of the partition. To escape from getting stuck at a local minimum we can use r random starts. Specifically, we can repeat the process r times and select the final clustering with the minimum SSE from the r runs [235, 236].

We keep clustering until we have a total of n_w sub-clusters, each of which contains the set of descriptors closest to its center. We call each of these sub-clusters a word. This clustering can be interpreted as a vocabulary tree, where k corresponds to the branching factor at each level, and each word corresponds to a path from root to leaf. Figure 3.44 illustrates this equivalence. Note that we keep track of the image corresponding to each descriptor.

At the end of the process each image i can be represented as an n_w dimensional vector d_i, where n_w is the total number of words in the tree. At each index $j \in [1, n_w]$ in d_i an entropy weighting [196] is applied so that

$$d_i[j] = \frac{N_j^i \ln \frac{M}{M_j}}{N_i}, \tag{3.36}$$

where N_j^i is the the number of descriptors of the i-th database image associated with the j-th word, M is the total number of database images, M_j is the number of database images with at least one descriptor belonging to the j-th word, and N_i is the total number of descriptors found on the i-th image. Based on the results of [196] we chose $k = 3$ and $n_w = 10,000$ to create our vocabulary tree.



(a) Step 1 (b) Step 2

(c) Step 3 (d) Step 4

Fig. 3.43.: Four Main Steps in k-Means.

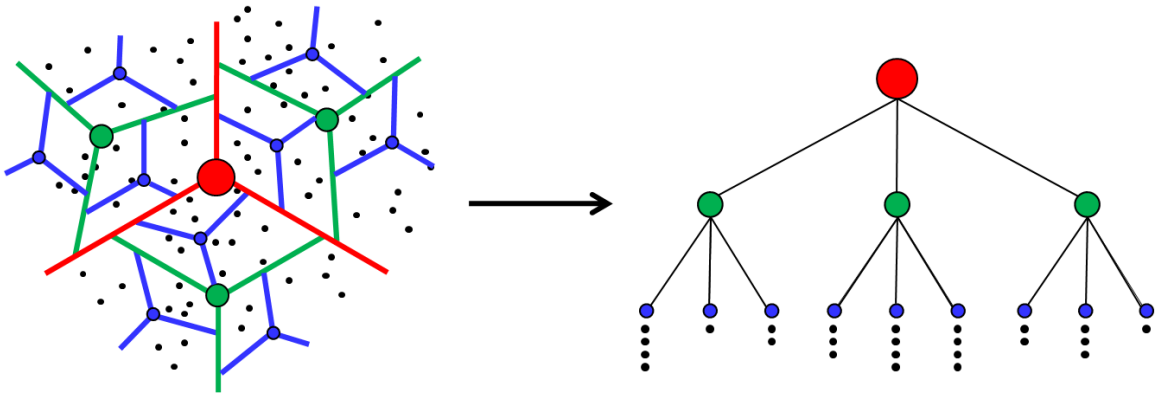


Fig. 3.44.: Vocabulary Tree Built From Hierarchical k-Means. Each black dot corresponds to a descriptor from a database image.

In order to match an input image I against an image in our database we first extract descriptors from I. Each of the input descriptors is pushed down the vocabulary tree to find its closest word and an n_w dimensional vector q is created following the same criteria explained above, such that

$$q[j] = \frac{N_j^q \ln \frac{M}{M_j}}{N_q}, \tag{3.37}$$

The method in [196] proposes a scoring method to find the closest match to I based on normalized differences, such that the closest match CM is

$$CM = \underset{i}{\operatorname{argmin}} \, ||q - d_i||_2^2 \tag{3.38}$$

However, in high-dimensional spaces (e.g. $n_w = 10,000$) the Euclidean distance exhibits properties of the phenomenon known as curse of dimensionality [237, 238]. The estimate of CM can be very poor if "boundary effects" are not taken into account. The boundary effect shows how the query region (i.e. a sphere whose center is the query point) is mainly outside the hyper-cubic data space. One way of illustrating this effect is to compare the volume ratio between a hypersphere with and a hypercube [239, 240]. The volume of a hypersphere with radius r and dimension d is

$$V_{hs} = \frac{2r^d \pi^{d/2}}{\Gamma(d/2)}, \tag{3.39}$$

where $\Gamma()$ is the Gamma function defined as

$$\Gamma(m) = 2 \int_0^\infty e^{-r^2} r^{2m-1} dr. \tag{3.40}$$

The volume of a hypercube with radius r and dimension d is

$$V_{hc} = (2r)^d. \tag{3.41}$$

Therefore, it can be seen that

$$\lim_{d\to\infty} \frac{V_{hs}}{V_{hc}} = \lim_{d\to\infty} \frac{\pi^{d/2}}{2^{d-1} d\Gamma(d/2)} = 0. \tag{3.42}$$

This shows how nearly all the high-dimensional space is contained in the "corners" of the hypercube.

Note that most average-case analyses of nearest neighbor searching techniques are made under the simplifying assumption that d is fixed and that the number of descriptors is so large relative to d that the boundary effects can be ignored. In Gang Graffiti Scene Recognition we find hundreds of high-dimensional descriptors from an input image, so we can use this assumption. However, in Gang Graffiti Component Classification we only extract dozens of high-dimensional descriptors, and making this assumption can be dangerous. Instead, we propose a majority voting matching approach, where CM is computed as

$$CM = \underset{i}{\operatorname{argmax}} \sum_{j=1}^{n_w} (N_j^q)^i, \tag{3.43}$$

where $(N_j^q)^i$ is the number of descriptors from q associated with the j-th leaf that match the i-th database image. Figure 3.46 illustrates the majority voting matching approach. Note that a drawback of the basic majority voting classification occurs when the class distribution is skewed. That is, samples of a more frequent class (i.e. graffiti component) tend to dominate the prediction of the query [241]. Therefore we need to make sure that the training data for Gang Graffiti Component Classification contains the same number of samples for each class.

The main advantage of using a vocabulary tree for image retrieval is that its leaves define the quantization, thus making the comparison dramatically less expensive than

Table 3.11: Parameters and thresholds used in Content Based Gang Graffiti Image Retrieval.

Parameter	Description	Value
k	Branching factor	3
n_w	Number of leaves	10,000

previous methods in the literature [196, 242, 243]. Also, once the vocabulary tree is built, new images can be added by just pushing down its descriptors.

The scalability of the vocabulary tree can be inferred from the results of [196], shown in Figure 3.45. The retrieval performance increases significantly with the number of leaf nodes, the branch factor, and the amount of training data.

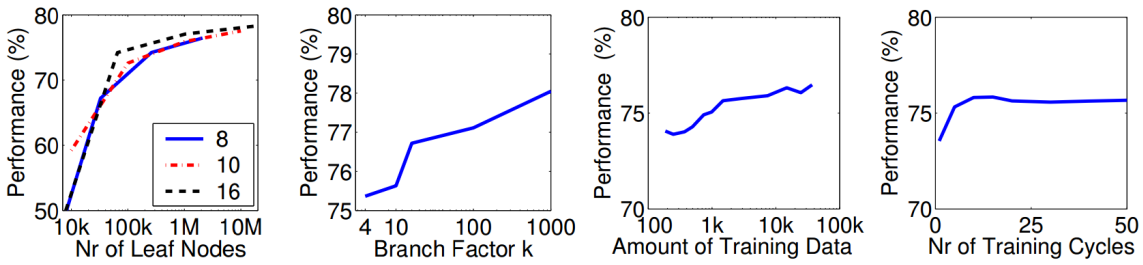


Fig. 3.45.: Scalability Results of Vocabulary Tree tested on a 6,376 ground-truth image dataset [196]. From left to right: Performance vs number of leaf nodes with branch factor k = 8, 10 and 16. Performance vs k for one million leaves. Performance vs training data volume in 720×480 frames, run with 20 training cycles and $k = 10$. Performance vs number of training cycles run on 7K frames of training data and $k = 10$. The image belongs to [196].

Currently, SIFT features are used for both Gang Graffiti Scene Recognition and Gang Graffiti Component Classification. However, note that the k-means clustering approach accepts any type multi-dimensional vector.

Table 3.11 shows all the parameters/thresholds we used including empirically derived parameters.

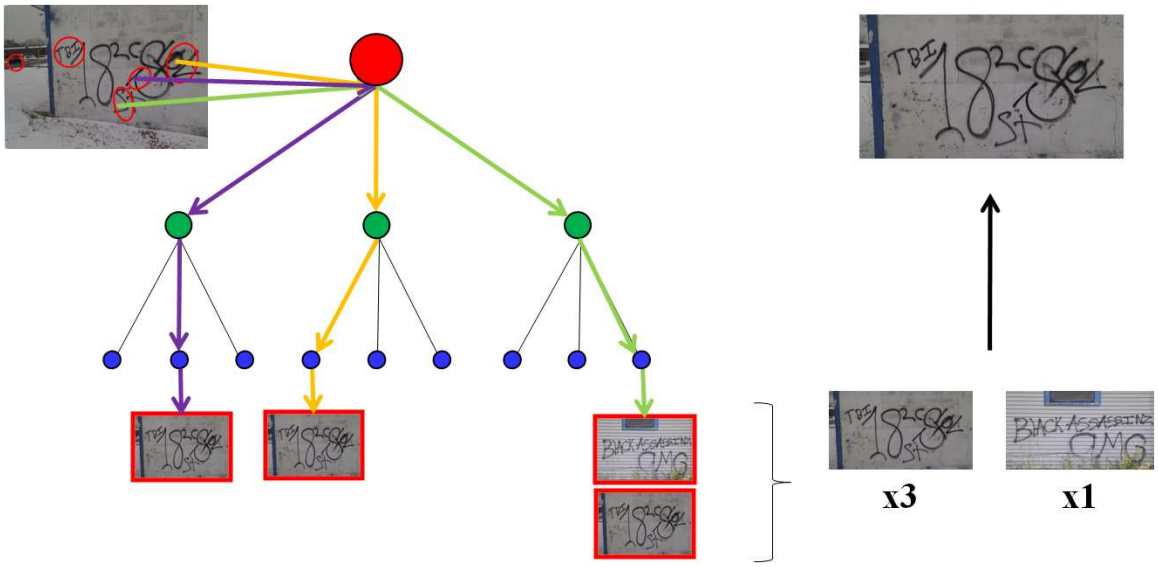


Fig. 3.46.: Majority Voting Matching.

3.8 System Implementation

3.8.1 System Architecture

We implemented the "mobile" part of the GARI system as an application for Android and iOS devices. We also have a web-based interface accessible from any web browser. Figure 3.47 illustrates the GARI system, which is divided in two groups:

1. **Client-side**: Implemented operations on the mobile device and communicate with the database (server) of gang graffiti through either WiFi or 4G/3G networks.

2. **Server-side**: Implemented operations on the database of gang graffiti and communicate with the client.

The client-side includes the device and methods available to the users, either to operate without the use of a network connection (offline services) or to make queries to the database (online services). The offline services are only available from Android devices (Section 3.8.3). The online services are available from both Android devices or any web browser (e.g., Internet Explorer, Mozilla Firefox, Google Chrome). This includes desktop and laptop computers as well as Blackberry smartphones (Section 3.8.4). The server-side includes all operations done on the server, including image analysis and queries to the database from both the Android application and the web-based interface. The database comprises gang graffiti images and metadata information for each entry, such as EXIF data, image geolocation and the results of the image analysis on each image whether it was done on the server or client.

3.8.2 GARI Databases

In this section we describe how the image database is organized. We will first describe the database schema and then show by an example how the information

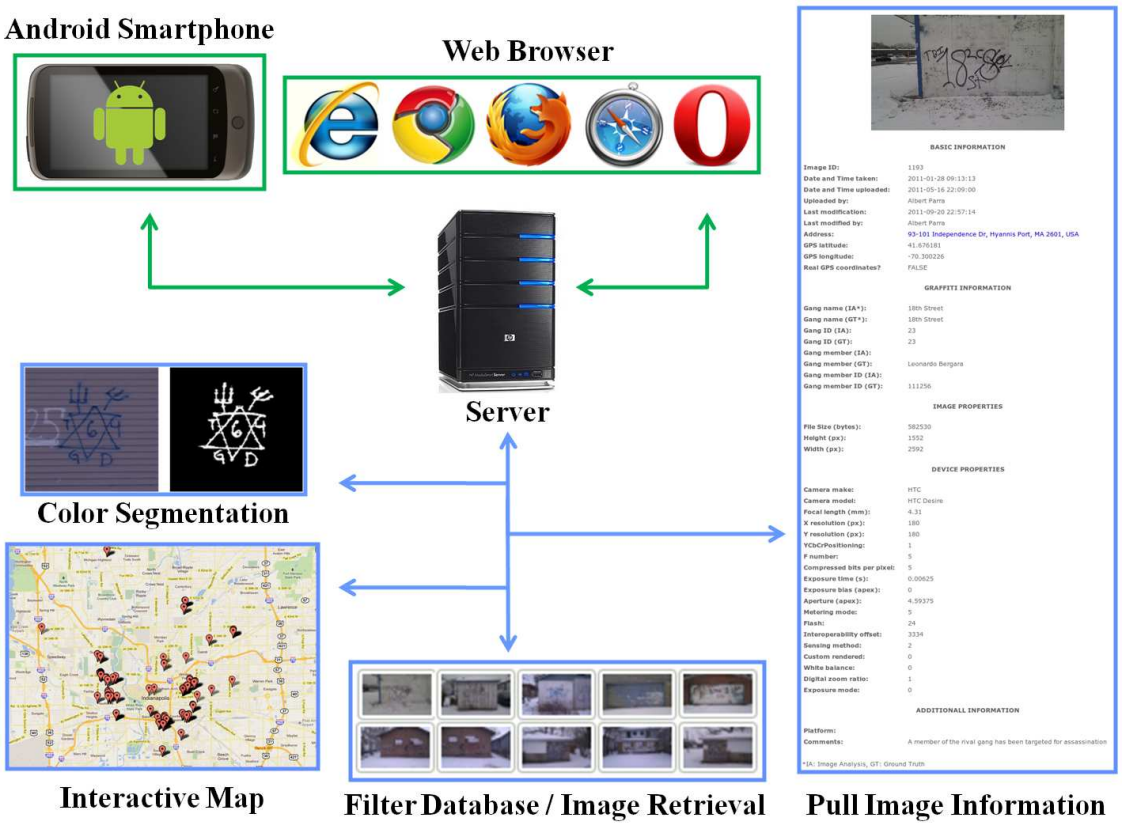


Fig. 3.47.: Overview of The GARI System - Client-Side Components (green) and Server-Side Components (blue).

GARI acquires is added to the database. The database of gang graffiti was deployed for three uses:

1. To collect and organize graffiti images acquired by first responders. This includes the images, metadata, and any interpretation or other information provided by the first responder.

2. To store the results of the image analysis.

3. To manage first responders' credentials, allowing them to access the services available through the Android/iOS applications and the web based interface.

Our database is implemented in PostgreSQL [244] on a Linux server. It consists of eight tables structured as shown in Figure 3.48. Note that the schema does not show all the fields in all the tables but just the relevant fields to indicate the association between the tables. Also the various IDs mentioned below (e.g. image ID) will be discussed in more detail after the tables are described in the following list.

1. **images**: Stores EXIF data from the images along with image location and general image information and the results from the image analysis. The fields related to this table are shown in Tables D.1, D.2, D.3 and D.4 in Appendix D.

2. **imageColors**: Stores all color IDs related to each image ID. This table is especially useful when more than one color is found in the same graffiti image.

3. **colors**: stores the relationship between color IDs and color names.

4. **imageBlobs**: Stores the number of blobs in each graffiti, the ID of each graffiti component for each blob, and the color ID of each graffiti component. This also stores special attributes of graffiti components. These attributes may include a specific graffiti component being crossed-out, upside-down, etc. Table D.6 in Appendix D describes the fields of this table.

5. **blobComponents**: stores the relationship between graffiti component IDs and graffiti component names, as well as the type ID for each graffiti component. Each graffiti component belongs to any of the following types: symbol, character, number, acronym, nickname, string.

6. **componentTypes**: stores the relationship between type IDs and type names.

7. **gangComponents**: stores the relationship between gang IDs and gang names, as well as the graffiti component ID (or multiple graffiti component IDs) associated with each gang. This table is especially useful when more that one graffiti component is associated with the same gang name.

8. **users**: Stores users' credentials to access to the system services as well as information concerning administrative privileges, email addresses, and registration and login status. Table D.5 in Appendix D describes the fields of this table.

Note that currently we only populate the tables **images** and **users**. The database relationships between all the tables are implemented and are ready to be used in the future (see Section 6).

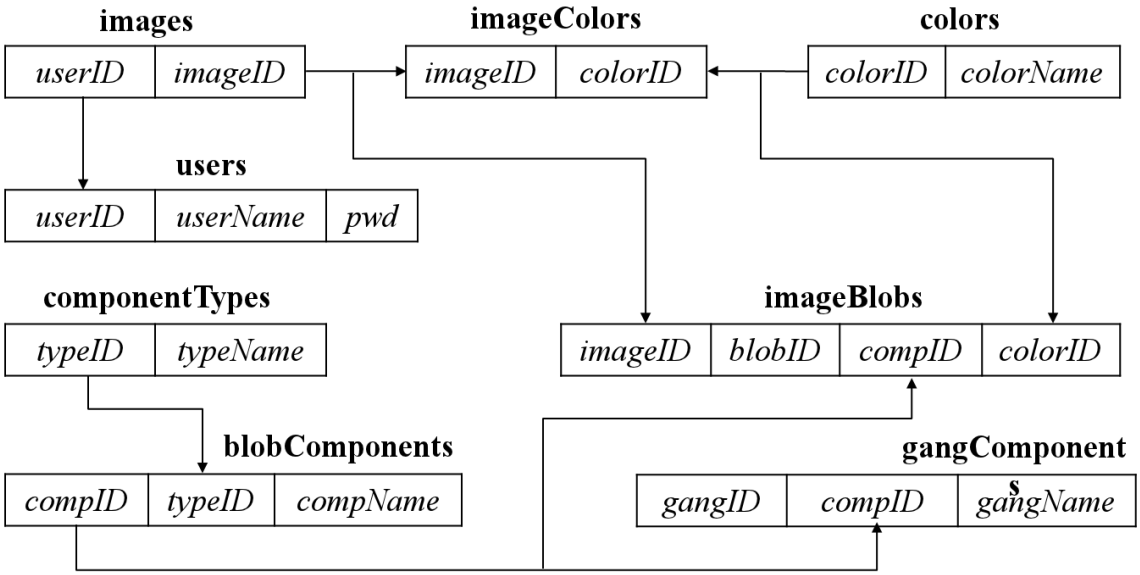


Fig. 3.48.: Database Schema Showing The Associations Between the Tables in the Database.

Adding Images to the Database

The following example illustrates the process of adding a graffiti image to the database. The image analysis is assumed to have been completed. Figure 3.49 shows the example image that has been manually labeled to facilitate the explanation. Each labeled circle represents a blob and each blob contains a distinguishable graffiti component. The blob labeling of the image corresponds with the field *blobID* from table *imageBlobs* in the database.

First, we fill table *imageColors* with the colors found in the graffiti. This is, black, green, and blue. Second, we analyze the blobs separately:

1. Color: black. Graffiti component: X3.
2. Color: green. Graffiti component: SPV.
3. Color: blue. Graffiti component: X3.
4. Color: blue. Graffiti component: LK. Crossed-out in green.
5. Color: blue. Graffiti component: ES. Crossed-out in green.

Note that the meaning of the acronyms and the type of the graffiti components is not addressed here. This information is assumed to already exist in the database.

Once the image analysis is complete the image, along with the blob information, is added to the database. Figure 3.50 shows the database fields filled with the information obtained from the graffiti in Figure 3.49. First, the user ID of the first responder who captured the image and the image ID are added to the *images* table. The image ID is a unique identifier of the graffiti image and it is automatically updated every time an image is uploaded to the server. Although it is not shown in Figure 3.50, some additional image information (i.e., EXIF data, GPS coordinates) is extracted from the uploaded image and added to the *images* table. Second, the color IDs for the three colors found in the graffiti, which are obtained by checking the color description field, (labeled *colorName* in Figure 3.50), are added to the *imageColors* table, and

linked to the graffiti ID. At the same time, the five blobs are added to the *imageBlobs* table. Each blob has a corresponding graffiti component ID, which is obtained by checking the graffiti component description field, (labeled *compName* in Figure 3.50), of the *blobComponents* table. Each graffiti component has a color associated with it and can activate one or many attributes in the same table (see Table D.6 for all the attributes). In this example, blobs one to three do not have any additional attribute. Blobs four and five have activated the crossed-out attribute.

Note that this process is totally objective. That is, the information uploaded to the database does not require any interpretation from the first responder. With all the objective information available in the tables and the associations between the data one can produce an informed graffiti interpretation. For example, we have added graffiti components with IDs 27 (SPV) and 29 (LK). These IDs are associated with specific gang names in the *gangComponents* table. The same reasoning could be used if the graffiti did not contain any specific content with just the graffiti color being identified. Additional tables can relate gang IDs with color IDs effectively providing the results of gangs matching the specific color or colors.

Fig. 3.49.: Example of Graffiti (Manually Labeled).

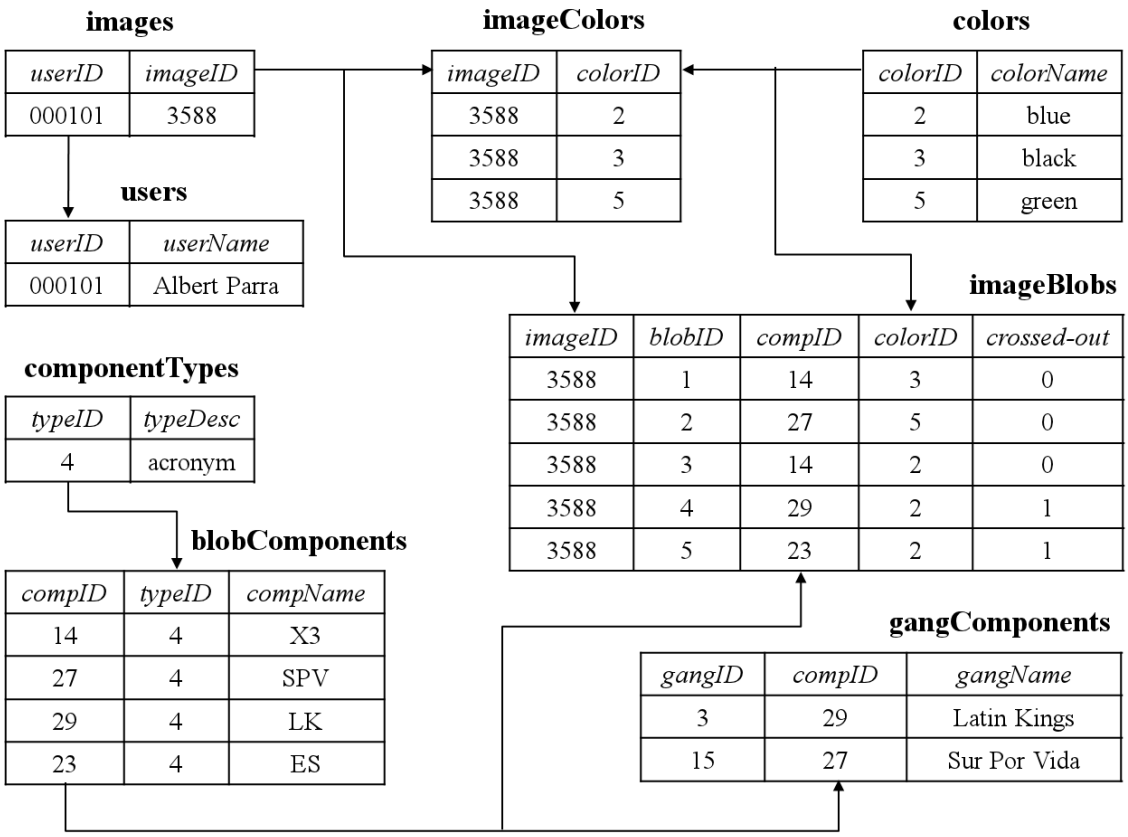


Fig. 3.50.: Database Fields With Information From The Graffiti in Figure 3.49.

3.8.3 Android/iOS Implementation

We implemented the GARI system on Android and iOS devices as summarized in Figure 3.51. We called this application Mobile GARI. In this section we describe how the application works and describe its user interface.

Overview

A user takes an image of the gang graffiti using the embedded camera on the device via the Graphical User Interface (GUI). The EXIF data of the image, including GPS location and date and time of capture, is automatically added to the image header.

The user can then choose to upload the image to the server to be included in the database of gang graffiti, find similar images in the database of gang graffiti, or do color recognition. The first option, uploading to the server, allows the user to send the image and the EXIF data to the server creating a new entry in the database. The second option, find similar images, allows the user to send the image to the server and find gang graffiti images that match part or all of the contents of the image. The third option, color recognition, allows the user to trace a path in the current image using the device's touchscreen. The color in the path is then automatically detected (Section 3.4) and the result is shown to the user. The database of gang graffiti can then be queried to retrieve graffiti images of the same color.

Another option is to browse the database of gang graffiti given various parameters such as the distance from current location or date and time. The thumbnail images that match the query are downloaded from the server and shown to the user on the mobile telephone. The user can then browse the results to obtain more information about the specific graffiti. Note that in order to browse the database of gang graffiti a network connection is required.

We implemented the system on different smartphones makes and models, but always targeting version 3.2 of the Android operating system (OS). We chose Android OS version 3.2 to cover as much user market as possible while still being able to include the necessary features. Since Android applications are generally forward-compatible with new versions of the Android platform, by choosing OS version 3.2 we cover 78.7% of the market (as of March 2014) [245].

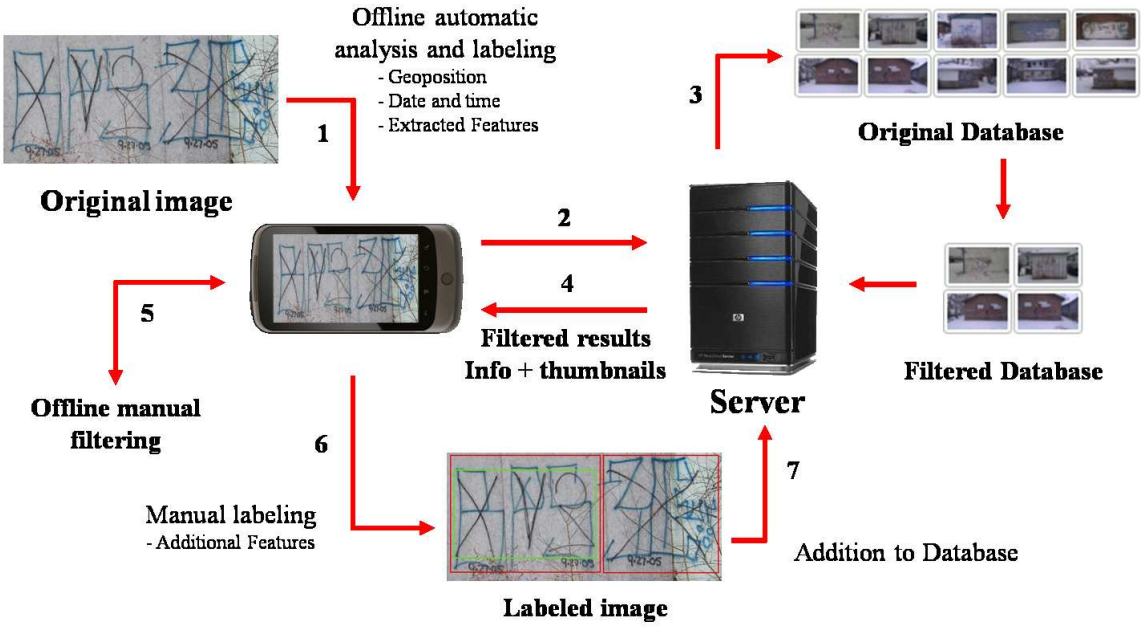


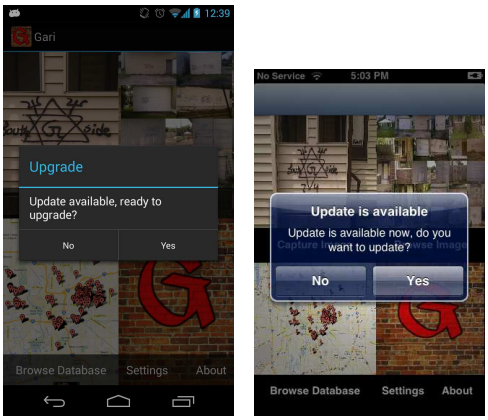
Fig. 3.51.: Overview of the GARI System.

User Interface

Our Android application does not require the use of a network connection. However it is mandatory if the user wants to browse the graffiti database or upload images to the graffiti database. The application automatically checks for updates when launched, notifying the user if a new version is available (Figure 4.25). A user must be assigned a User ID (equivalent to a First Responder ID) and a unique password in order to use GARI. Once the User ID and password has been entered, the main screen is presented. The menu options are displayed on the main screen (Figure 4.26a/3.53c) and on the secondary screen (Figure 4.26b/4.26b) when an image is captured or browsed. In Android devices, the menu button brings additional options when available. Note that the menu button can be a hardware key (Figure 3.54a) or a software key (Figure 3.54b) depending on the device used. In iOS devices, the

additional options are presented on the screen as buttons. The main screen includes the following options:

- Browse Image
- Browse Database
- Capture Image
- Send to Server (available after browsing or capturing an image)
- Analyze Image (available after browsing or capturing an image)
- Settings
- About

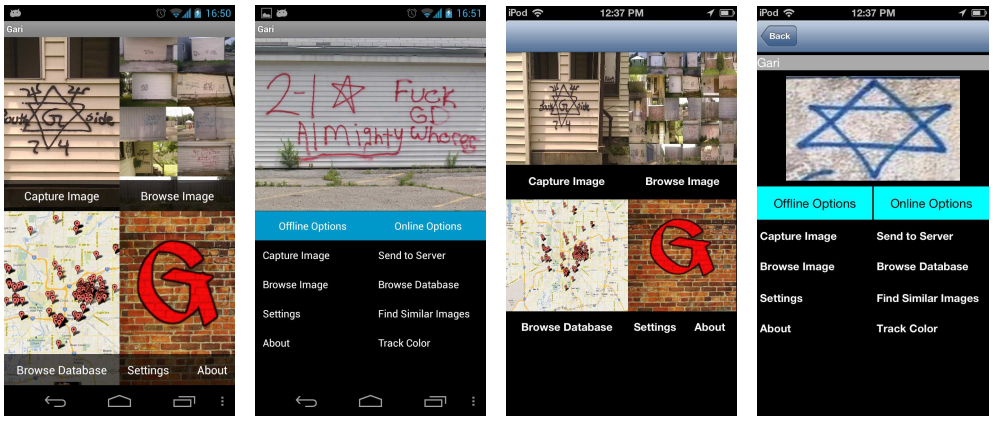


(a) Android (b) iPhone

Fig. 3.52.: Automatic updates.

Browse Image

The user has the option to browse images stored on the Android device, to later upload them to the server or analyze them. Note that the entire phone image gallery is browsed, including images that have not been taken using the Mobile GARI application. When the option "Browse Image" is tapped, a directory browsing window is

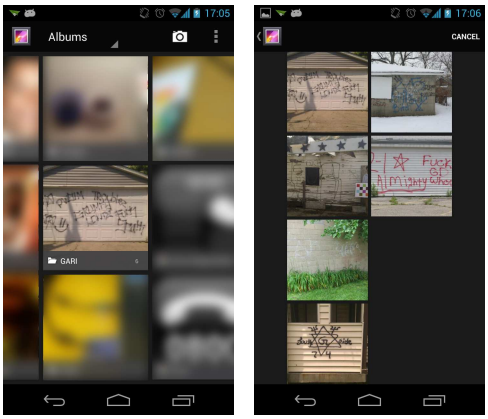


(a) (b) (c) (d)

Fig. 3.53.: User options screens for Android (4.26a, 4.26b) and iPhone (3.53c, 3.53d).

(a) Hardware key (b) Software key

Fig. 3.54.: Examples of location of the menu button (red square) on Android devices.



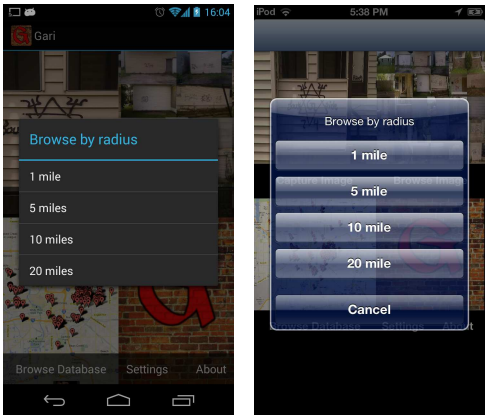
(a) (b)

Fig. 3.55.: Example of image browsing.

opened, and the user can search and select the desired image. Figure 3.62 shows an example of browsing.

Browse Database

The menu option "Browse Database" allows the user to browse the database by radius. That is, it extracts from the database all the images in a given radius from the current location. Figure 3.56 shows the dialog where the user can select a radius between 1 mile and 20 miles.



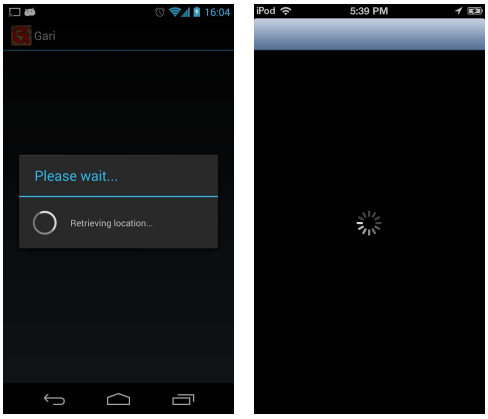
(a) (b)

Fig. 3.56.: Browse by radius screen for Android (left) and iPhone (right).

When a specific radius is chosen, the application checks for the device location automatically, in order to add the GPS coordinates to the image. Depending on the system used (Network (3G/4G or WiFi) or GPS), it can take up to 30 seconds to acquire the location. The user is notified during the period, as shown in Figure 3.57.

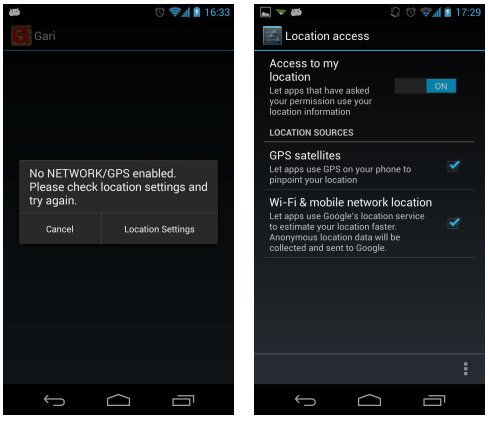
In Android devices, if the location system is not enabled on the device, the user is notified and taken to the location settings (Figure 3.58), where the location systems can be enabled.

Once the location is locked, the application contacts the image database and checks how many thumbnails have to be downloaded (Figure 3.59a/3.59c). If the user accepts, the information that matches the query is retrieved (Figure 3.59b/3.59d). Figure 3.60 shows an example of the results, where each line contains a thumbnail of



(a) (b)

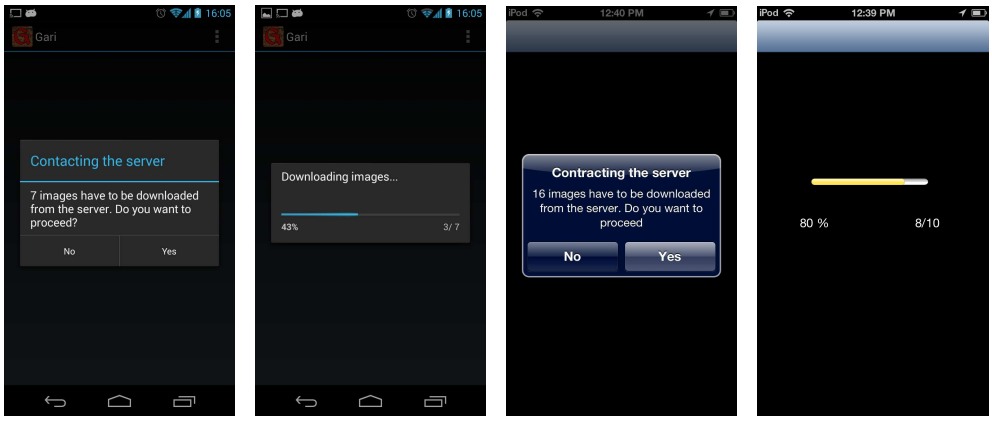
Fig. 3.57.: Progress dialog notifying the user of a location retrieval, for Android (left) and iPhone (right).



(a) (b)

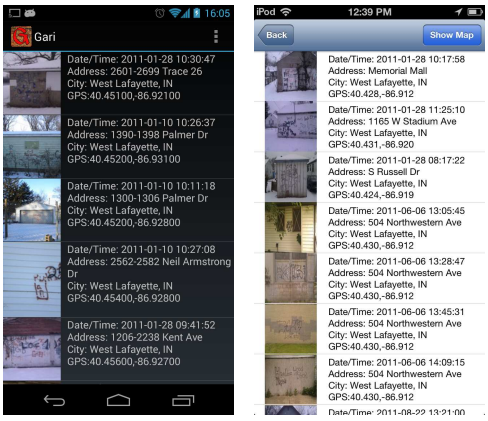
Fig. 3.58.: 3.58a Dialog notifying the user that no Network or GPS systems are enabled, and 3.58b location settings of the device, for Android.

a graffiti or tattoo and basic information about it, including the date and time the image was taken, and its GPS latitude and longitude.



(a) (b) (c) (d)

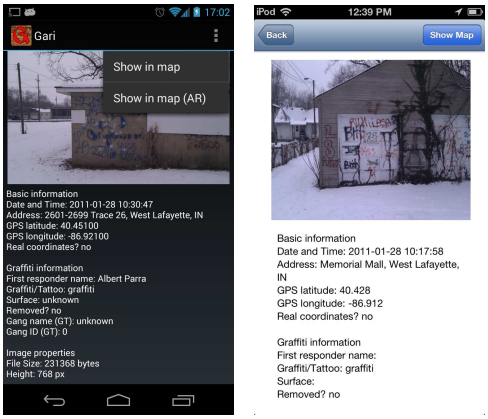
Fig. 3.59.: Screen notifications during database browsing for Anroid (3.59a, 3.59b) and iPhone (3.59c, 3.59d).



(a) (b)

Fig. 3.60.: Results after querying the image database for Android (left)) and iPhone (right).

To obtain more information about a particular graffiti or tattoo, the user can tap on either the thumbnail or the text field, and the application will contact the server, extracting a larger image and the information available. Figure 3.61 shows an example of the extended results. The text field includes information about the

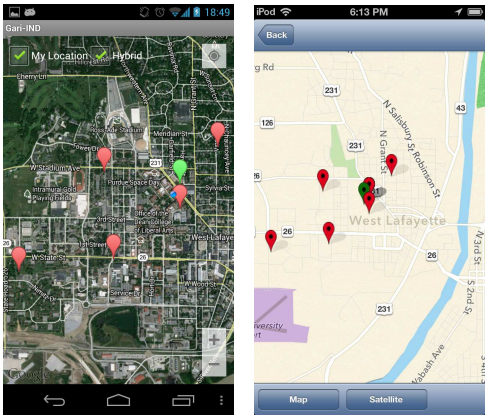


(a) (b)

Fig. 3.61.: Extended results after querying the image database for Android (left) and iPhone (right).

graffiti or tattoo taken from the database tables. The available fields are detailed in Appendix D.

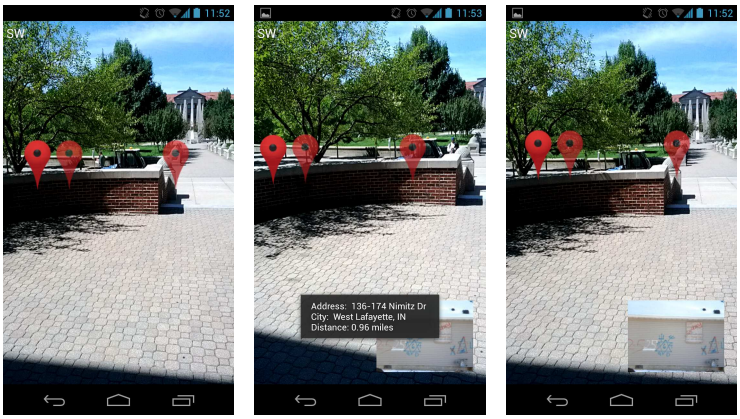
Whether the user is in the results view or in the extended results view, the menu key will have the option "Show in map." It allows the user to display the position of multiple graffiti or tattoo or focus on a single image (green marker on Figure 3.62a/3.62b), depending on the current layout. In Android phones the user can choose to display his/her current location via the "My Location" option, and switch between normal and hybrid maps via the "Hybrid" option. In iOS devices the user can switch between normal and hybrid using the buttons on the map.



(a) (b)

Fig. 3.62.: Graffiti locations displayed on a map for Android (left) and iPhone (right)

Similar to the "Show in map" option, the menu key will have to option "Show in map (AR)." AR stands for Augmented Reality. It allows the user to display the position of graffiti and tattoo locations on top of the camera feed on the mobile phone. Figure 3.63 shows an example. As the user moves the mobile phone around, the screen gets updated and shows graffiti and tattoo locations in the camera range as pins. When tapping on a pin, a dialog appears at the bottom displaying the address, city and distance of the graffiti/tattoo from the mobile phone. Also, the image thumbnail is shown in the bottom right. When tapped, the user is redirected to the extended results page (Figure 3.61).



(a) (b) (c)

Fig. 3.63.: Graffiti locations displayed on an Augmented Reality feed for Android

Capture Image

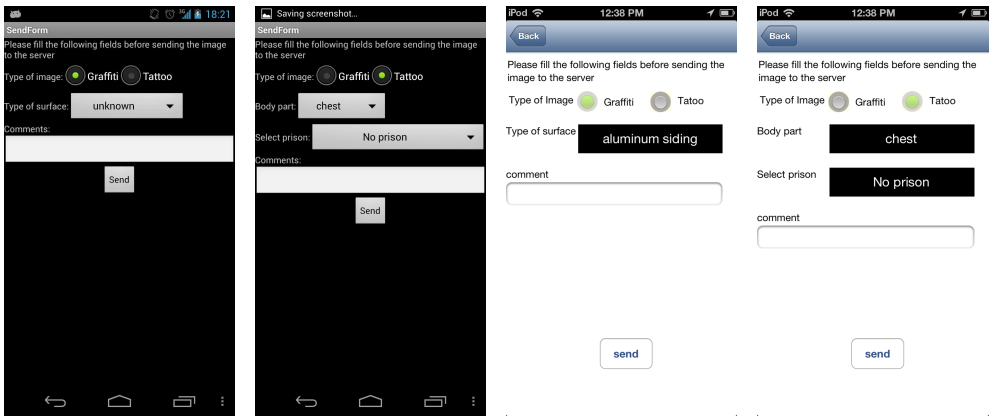
The menu option "Capture Image" starts the image acquisition. The user just has to point to the graffiti or tattoo and wait for the three second countdown followed by automatic image acquisition. The countdown is shown in the center of the screen, as illustrated in Figure 3.64. The countdown is automatically restarted if the smartphone registers a considerable amount of shaking, in order to minimize the risk of taking blurred images. After the image is automatically captured the application checks for motion blur and lack of illumination, and restarts the counter to take a new image if necessary. The application automatically checks the user's current location after acquiring an image.

Fig. 3.64.: Camera Activity.

Send to Server

The menu option "Send to Server" allows the user to send the current image to the server. First, the user will be prompted to select the source of the image, either graffiti or tattoo (Figures 3.65a/3.65c and 3.65b/3.65d). After tapping on "Send" the image is uploaded to the server on the background. While an image is being uploaded, the user can keep using the application and send more images. A queue will be automatically created and the images will be sent sequentially. If the Internet connection is lost, the application will wait until the connectivity is restored to restart the uploading process. If the application is closed or the mobile device is shut down during an upload, the file will be automatically uploaded next time the user launches the application. Figure 3.66 illustrates the process. An icon on the notification bar (top of the screen) shows the status of the upload. By dragging down the notification

bar the user can see more information about the upload progress. If the image is successfully added to the database, the application will also extract the information uploaded, and will display it to the user (Figure 3.67).



(a) (b) (c) (d)

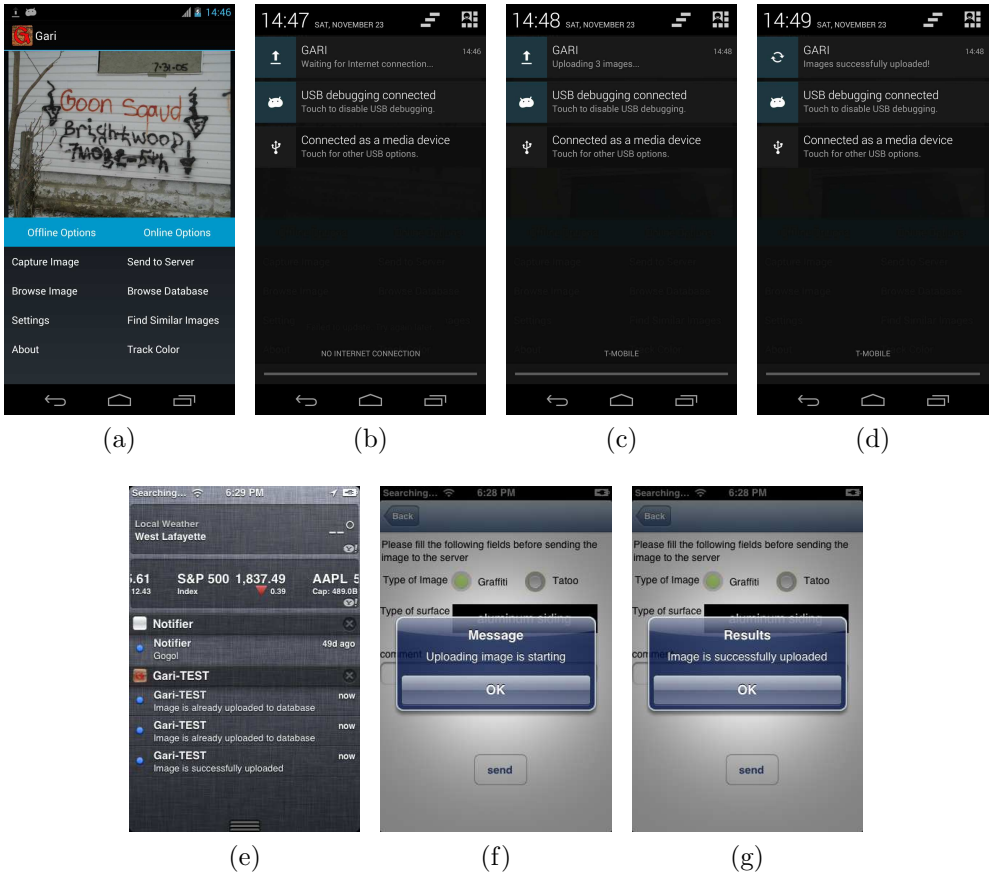
Fig. 3.65.: Result of uploading an image to the server for Android (3.65a and 3.65b) and iPhone (3.65c and 3.65d).

Find Similar Images

The menu option "Find Similar Images" allows the user to find similar images to the current image being displayed on the secondary screen. The image is sent to the server and analyzed. When the analysis is done, the server sends back a list of matching candidates. Figure 3.68 shows the process. The options for this list are the same as the ones described for the results from browsing the database. Note that the matching candidates in the list are sorted by score, where the first entry corresponds to the most similar image to the query.

Analyze Image

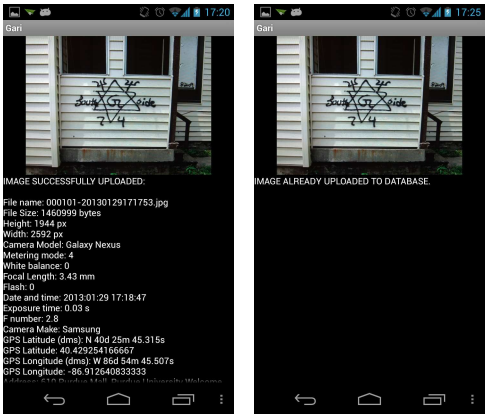
The menu option "Analyze Image" allows the user to aid the application in detecting the gang graffiti components. This option is only enabled once an image has been captured or browsed. First, the user has to select a region of the image containing the graffiti color, as shown in Figure 3.69a/3.69c. When the desired area is selected and "Save" is tapped, the user can create a path on the image using their finger, as shown



(a) (b) (c) (d)

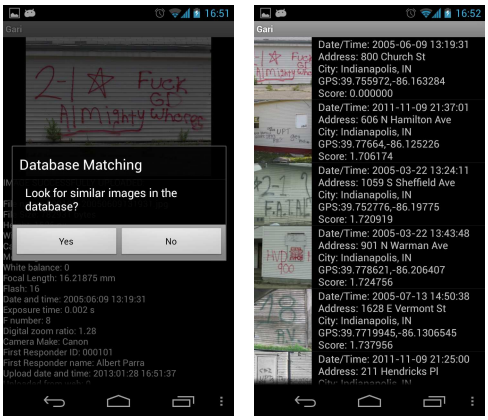
(e) (f) (g)

Fig. 3.66.: Image uploading on the background on Android (top) and iPhone (bottom). From left to right (Android): Uploading image (icon), waiting for Internet connection, uploading 3 images, image successfully uploaded. From left to right (iPhone): Messages on the notification bar, Uploading image (message), image successfully uploaded (message).



(a) (b)

Fig. 3.67.: Image upload successfully (3.67a) and image already uploaded to database (3.67b).



(a) (b)

Fig. 3.68.: Screen notifications when finding similar images (Android).

in Figure 3.69b/3.69d. There is no need to trace the entire content of the area with the same color. Just a significant sample is enough to determine the color. Figure 3.69b/3.69d also shows the available options. In Android devices the "Undo" option removes the last path created; the "Clear" option clears all the paths created; and the "Analyze" option obtains the current path and analyzes the color. In iOS devices the "Analysis" option obtains the current path and analyzes the color. The image and the recognized color are then sent to the server for analysis, and the results are given back to the user as a list of thumbnails, classification results and gang graffiti colors, as shown in Figure 3.70.

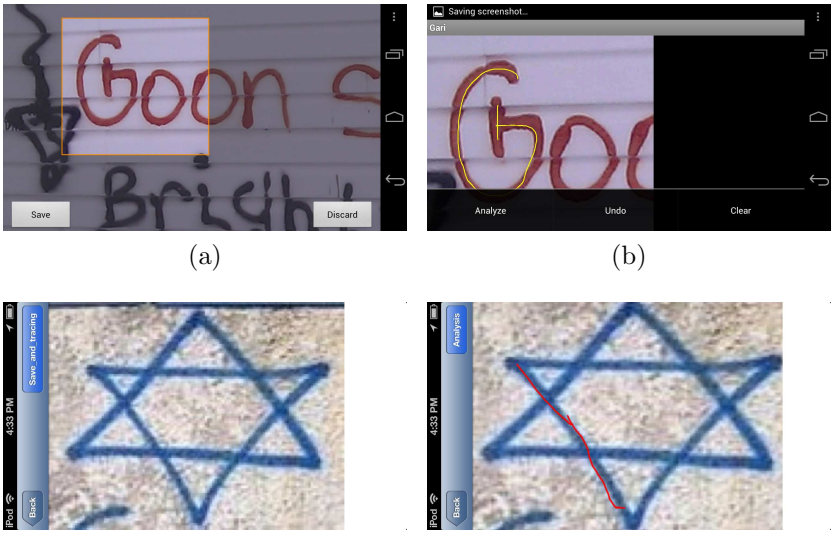


Fig. 3.69.: Steps to follow when selecting the region to analyze the color for Android (top) and iPhone (bottom).

Figure 3.71 shows the result of the color tracing. The application then extracts from the database all the gangs that match the detected color. There is also the option "Browse database by color", which queries the database and extracts all the images in the database that match the traced color. Figure 3.71b shows an example. After color recognition the user can send the image to the server for automatic graffiti analysis.

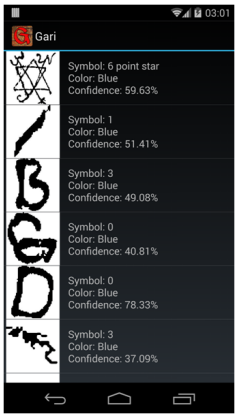
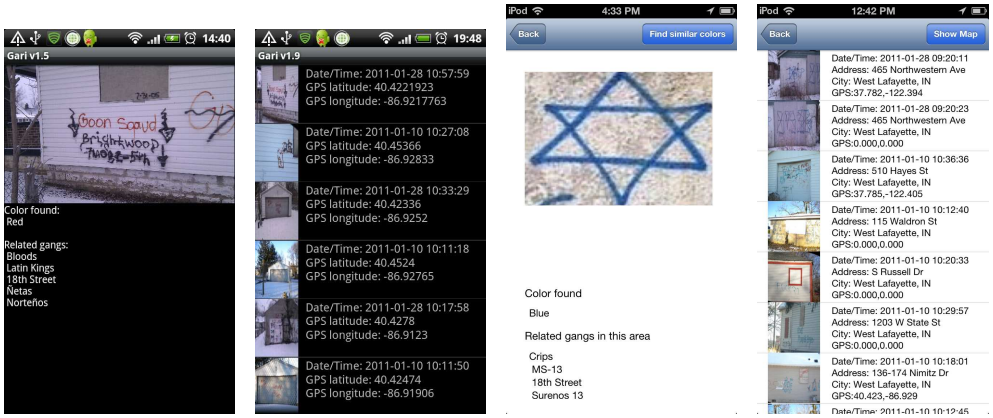


Fig. 3.70.: Image Analysis Results.



(a) (b) (c) (d)

Fig. 3.71.: Gangs related to the traced color and images in the database that match the traced color for Android (3.71a, 3.71b) and iPhone (3.71c, 3.71d).

Security

Our Android application is used by first responders from multiple agencies. Therefore, it is mandatory to ensure that only authorized users can access and use the application. The connections to the server must be secure and all the information transmitted to and from the server must be encrypted (using the SSL/TLS protocol). The user credentials are sent every time the application contacts the server to make sure the connection is made by an authorized user. In the Android version we use ProGuard [246], a code optimizer and obfuscator for the Android SDK. It reduces the application size up to 70% and makes the source code more difficult to reverse engineer. It also improves the battery life by removing verbose logging code in a background service. An additional level of security includes the creation of two types of users:

- Regular users: Can switch between users, change their password, delete specific images only taken by themselves, and send crashlogs to the server.

- Administrative users: Can modify the server domain name/IP address, change user IDs, change passwords, delete specific images from any user, delete all images of any specific user, and send crashlogs to the server.

When launching the GARI application a dialog box automatically prompts the user for login credentials (Figure 4.33). The user is required to input a user ID and a password.

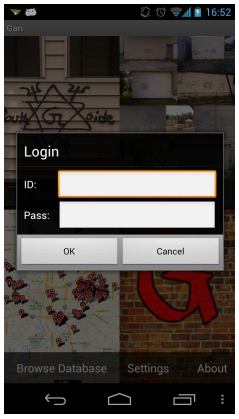


Fig. 3.72.: User ID Prompt.

The first time a user logs in the credentials are checked with the server and once they are validated they are stored in the device in an encrypted file. This allows the user to use the application without needing a network connection. Note that passwords are never stored as plaintext, neither on the device or the server. They are hashed using an MD5 cryptographic hash function [247]. We also use a login system in which the application creates a session for an authorized user that lasts 24 hours. After that period of time the user is required to login again.

All authorized users can access the "Settings" option from the main screen of the application. Figure 3.73 shows the various options. Note that no one can delete images from the server. At this time no one can edit the attributes of images retrieved from the server.

- Server domain/IP: the the address of the server to be changed by domain name or IP address (only available to administrative users).

- Switch user: allows one to open sessions for other users. Note that switching to another user ends the session for the current user.
- Change password: allows one to change the password used to access the application. Note that the password is changed for both the Android application and the web-based application.
- Send crashlog: allows system crash feedback to be sent to the server.

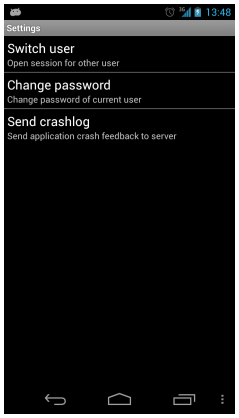


Fig. 3.73.: "Settings" Dialog, Showing the Various Options.

3.8.4 Web Interface

System Overview

We also implemented our system as a web interface that gives a user access to the graffiti in the database and provides the ability to upload, modify and browse most database contents as summarized in Figure 3.74. We called this application Desktop GARI. The user logs in into the "Archive" using authorized credentials. Note that the credentials are the same for both the Android application and the web services. The user can then either browse the database of gang graffiti or upload an image. If the choice is to browse the database, the user can check the graffiti images and their attributes or filter the database using parameters such as radius from a specific

location or address, capture data, upload data, or modified date. The results are shown as a list of thumbnail images with basic information that identifies the graffiti image. The user can then browse specific images and place them on a map, so to visually track gang activity. If the choice is to upload an image, the user can select a graffiti image from their local system (i.e., any device with a web browser). Some attributes can be adjusted through guided steps before adding the information to the database, such as location, gang information, or additional comments.

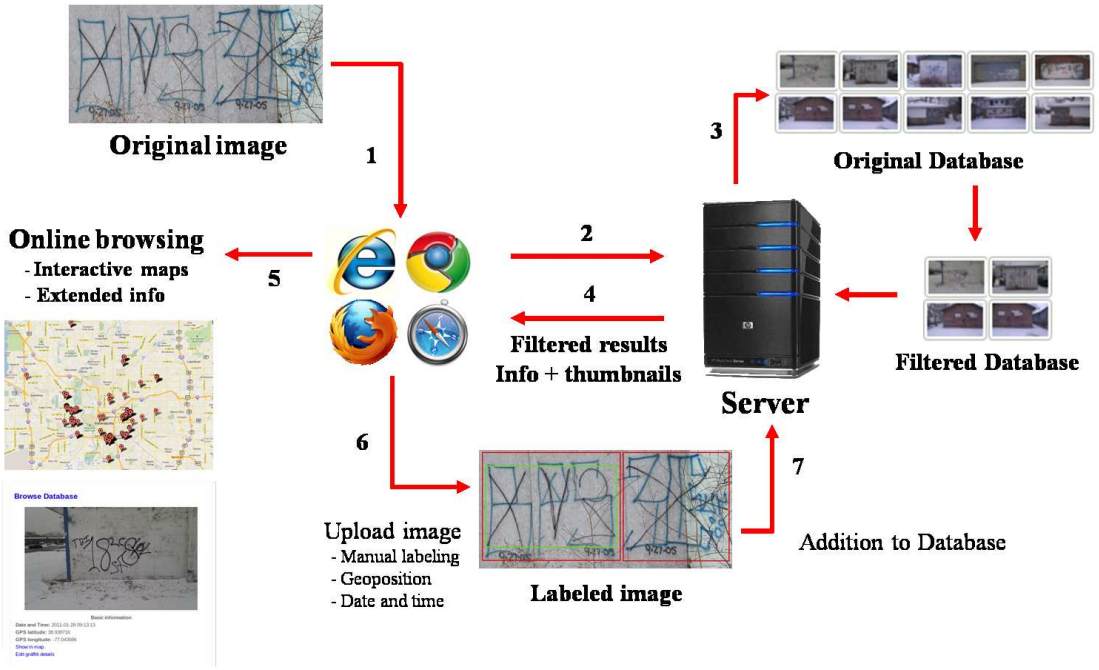


Fig. 3.74.: Overview of the Web Interface of the GARI System.

The web interface is available from any device with a web browser. This includes all desktop and laptop machines and all mobile telephones capable of browsing the web (e.g., iOS, Blackberry, Android devices). In some cases, the current location of the user is required in order to retrieve results from the database of gang graffiti such as when using the "radius" function to display graffiti on a map. Geolocation was introduced with HTML5 and it is widely implemented by many modern browsers.

However, only the latest browsers support this service. Table 3.12 lists the browsers and their support level for Geolocation.

Table 3.12: Web Browsers Supporting HTML5 Geolocation Service.

Browser	**Version**
Firefox	3.5+
Internet Explorer	9+
Google Chrome	5+
Safari	5+
iPhone Safari	+3.0 OS
Android	Through Gears API
Opera	10.6+

User Interface

As of March 2014 the GARI website is located at www.gang-graffiti.org. The main page contains information about the GARI project, its principal investigators, and the graduate students involved. Figure 3.75 shows a snapshot.

The "Archive" page (Figure 3.76) displays the options available a user. These include:

- Browse database
- Upload image
- Upload multiple images
- Create database report

A username and password is required to access the database contents. A user can use the same username and password used for the mobile application.

Browse database

The "Browse database" page (Figure 3.77) allows the user to either browse the entire database or to do a specific search. This includes:

- **Browse all database/graffiti/tattoo**: Retrieves from the database either images, only graffiti images, or tattoo images.
- **Search by radius**: Retrieves from the database all the graffiti and tattoos in a specific radius, from a specified location from the list. The locations in the list include the user's current location, the Video and Image Processing Laboratory (VIPER) at Purdue University, and the Indianapolis Metropolitan Police Department (IMPD). The "Current location" option requires the user to share their current location, as shown in Figure 3.78.
- **Search by Date**: Retrieves from the database the graffiti and tattoo images captured, uploaded or modified in a specific period of time.
- **Search by address**: Retrieves from the database the graffiti and tattoo images in a specific radius, from a specified address. Provides more flexibility than the "Search by radius" option.

The search results are shown in Figure 3.79. At first, only a small-scale image and basic information is displayed. Depending on the search various parameters are shown,including:

- Date/Time captured (uploaded, modified): date and time the image was acquired, uploaded or modified, depending on the search.
- Address: address where the image was acquired. A map showing the graffiti or tattoo location when clicked is available.
- More information: link to show additional information about the graffiti or tattoo.

- Image ID: image identifier in the database.
- Distance: distance from the user's current location to the graffiti or tattoo. Only available when searching by radius or address.

Each image or group of images can be displayed on an interactive map. Figure 3.80 shows an example of the interactive map when a single image is displayed. The image is placed on a map, and a balloon pops out, showing a thumbnail and some information about the image, including the date and time it was acquired, and its location in GPS coordinates. Figures 3.81 and 3.82 show an example of the interactive map when multiple images are displayed. Each marker represents the location of a graffiti or tattoo from the search results. From this map the user can click on any of the markers to see a thumbnail of the graffiti or tattoo, its location in GPS coordinates, and a link to obtain more information about the graffiti or tattoo. Figure 3.83 shows an example.

In the "More information" section, the user can see the information available in the database for a specific graffiti or tattoo. Figure 3.84 shows an example. The image can be clicked to enlarge it in a new window. Also, there are two additional options: "Show in map", and "Edit image details".

Upload Image

The "Upload image" feature (Figures 3.85 and 3.86) allows a user to upload an image to the database.

Once the image is uploaded, fields can be filled in by the user. These include:

- Assign GPS coordinates
  - By known address
  - By clicking on map
- User information
  - First responder name

  - First responder ID

- Graffiti/Tattoo information
  - Image Type
  - Surface type (if graffiti)
  - Body part (if tattoo)
  - Prison (if tattoo)

- Additional information
  - Gang name: from drop-down menu of known gangs or user's input
  - Gang member: gang member involved in the graffiti
  - Comments

Figures 3.87 and 3.88 show examples of filled fields adding information to the graffiti.

Clicking on "Submit Image" completes the editing and shows the user the final output of the image uploading session. Figure 3.84 is an example of this (the same information as clicking on "More information" when browsing the graffiti database).

Upload Multiple Images

The "Upload multiple images" feature (Figure 3.89) allows a user to upload multiple images to the database at the same time. By clicking on "Select files" the user can browse the computer to select one or multiple images to upload to the server (Figure 3.90). Multiple images can be selected using the SHIFT or CTRL buttons on the keyboard. By holding SHIFT when clicking on two files, it will select everything in between them. By holding CONTROL when clicking on files, it will select individual images. Once the images are selected a list of files to upload will be created as shown in Figure 3.91. By clicking on "Upload selected files" the images are uploaded to the server. As the images are being uploaded, the progress is shown to the user (Figure 3.92). Once all the images are uploaded, a preview screen is shown to the user, where basic information is automatically populated for each image (Figure 3.93). For each image, the user can populate the same fields as when using the feature "Upload image". After populating all the necessary fields, the user can click on "Submit images" located below the last image to update the information on the server. The results of the submission are shown as seen on Figure 3.79. Note that until the user clicks on "Submit images" no images are added to the database.

VIPER
Video and Image Processing Laboratory
VACCINE

Main
Archive

Gang Graffiti Recognition and Analysis Using a Mobile Telephone

Gangs are a serious threat to public safety throughout the United States. Gang members are continuously migrating from urban cities to suburban areas. They are responsible for an increasing percentage of crime and violence in many communities.

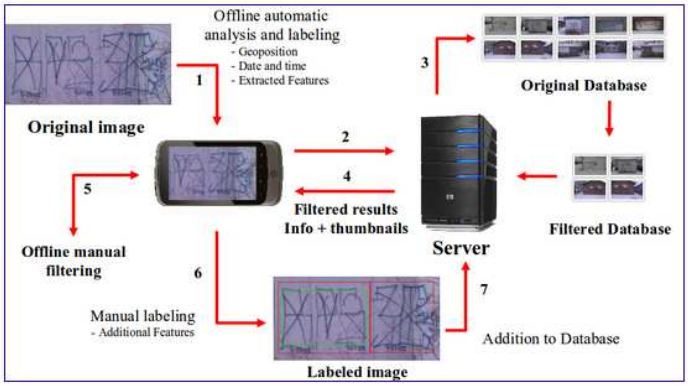
According to the National Gang Threat Assessment, approximately 1 million gang members belonging to more than 20,000 gangs were criminally active within all 50 states and the District of Columbia as of September 2008. Criminal gangs commit as much as 80 percent of the crime in many communities according to law enforcement officials throughout the nation.

Street gang graffiti is their most common way to communicate messages, including challenges, warnings or intimidation to rival gangs. It is, however, an excellent way to track gang affiliation and growth, or even sometimes to obtain membership information.

The goal of this project is to use the knowledge gained from our work in mobile devices and applications and leverage it towards the development of a mobile-based system capable of image analysis. This system will provide an accurate and useful output to a user based on a database of gang graffiti images.

The image analysis includes obtaining the metadata (geoposition, date and time) and extracting relevant features (e.g., color, shape) from the gang graffiti image. The information is sent to a server and compared against the graffiti image database. The matched results are sent back to the device where the user can then review the results and provide extra inputs to refine information. Once the graffiti is completely decoded and interpreted, it is labeled and added to the database.

This project is funded by the Department of Homeland Security's Visual Analytics for Command, Control and Interoperability Environments Center of Excellence (VACCINE) at Purdue University.



System Overview

Principal Investigators

Edward J. Delp, The Charles William Harrison Distinguished Professor of Electrical and Computer Engineering and Professor of Biomedical Engineering, Purdue University

Mireille Boutin, Assistant Professor of Electrical and Computer Engineering, Purdue University

Graduate Students

Albert Parra Pozo, Graduate Student, School of Electrical and Computer Engineering, Purdue University

Main | About | Publications | News

Last modified: Thu, 15 Sep 2011 00:27:51 EDT

Fig. 3.75.: Main Page of the Web Interface of GARI.

Fig. 3.76.: "Archive" Section of Desktop GARI.

Fig. 3.77.: "Browse database" section of the web-based interface for GARI.

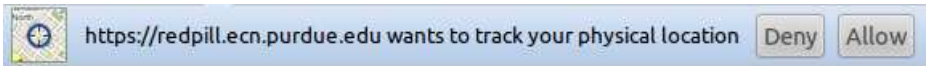


Fig. 3.78.: The current location of the user is only acquired upon request.

Fig. 3.79.: Results of browsing the database.

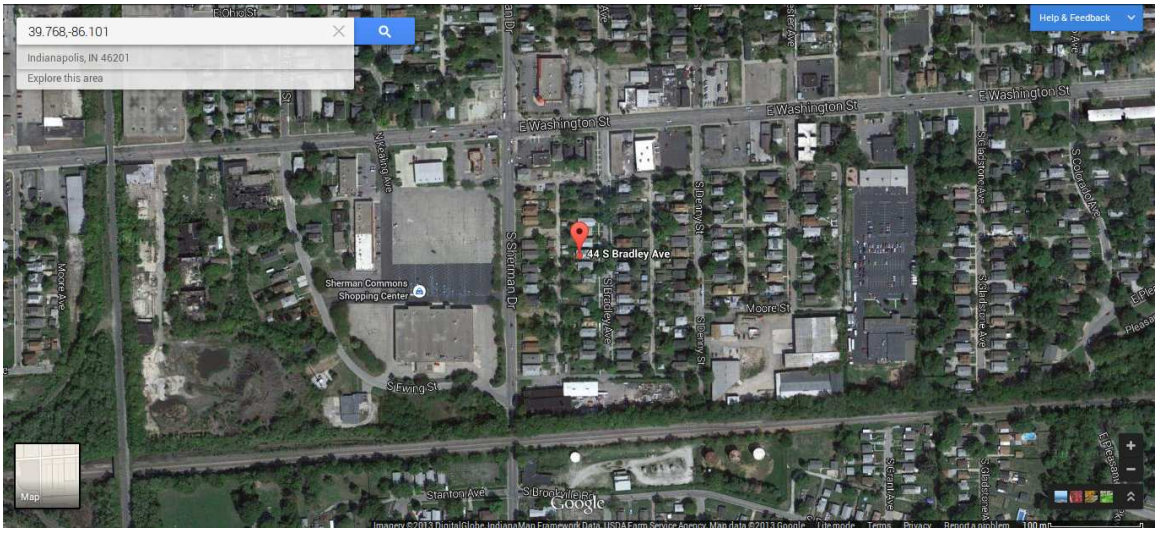


Fig. 3.80.: Example of the interactive map when a single image is displayed.

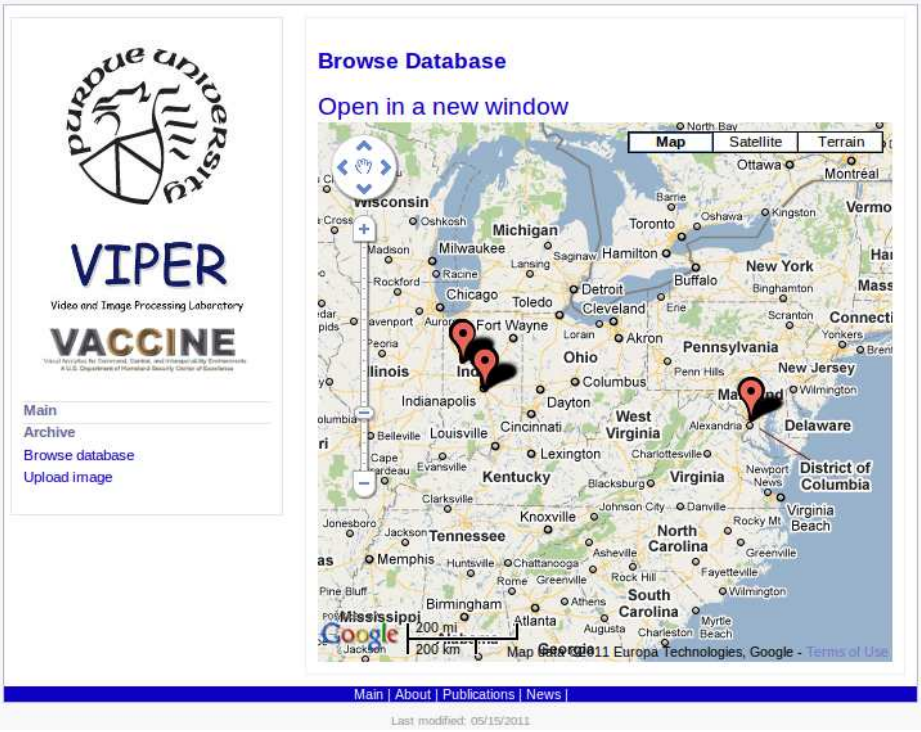


Fig. 3.81.: Example of the interactive map when multiple images are displayed.

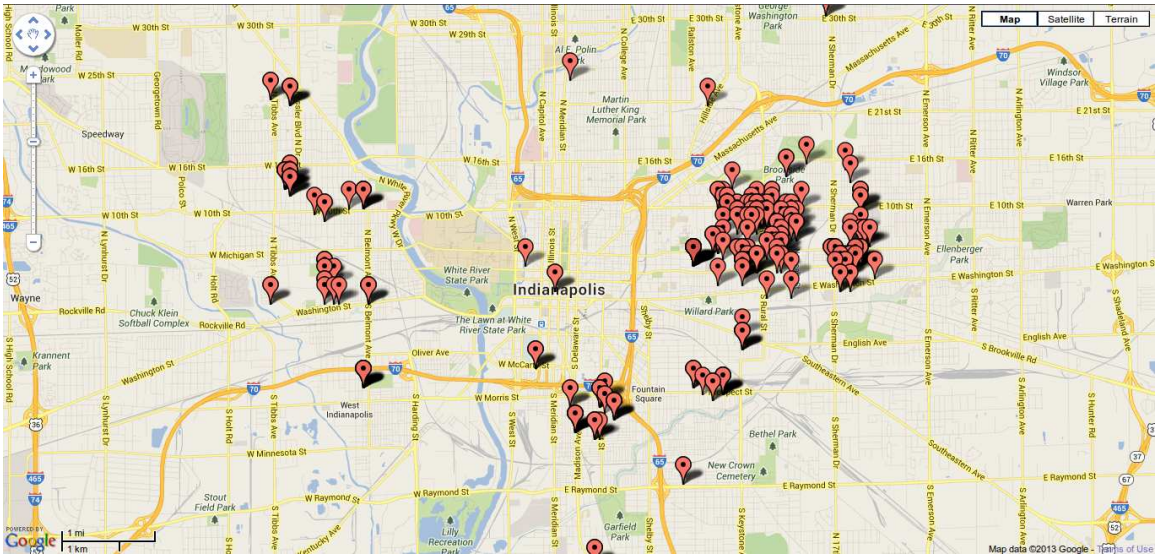


Fig. 3.82.: If "Open in a new window" is clicked, the interactive map expands to a full screen to make navigation easier.

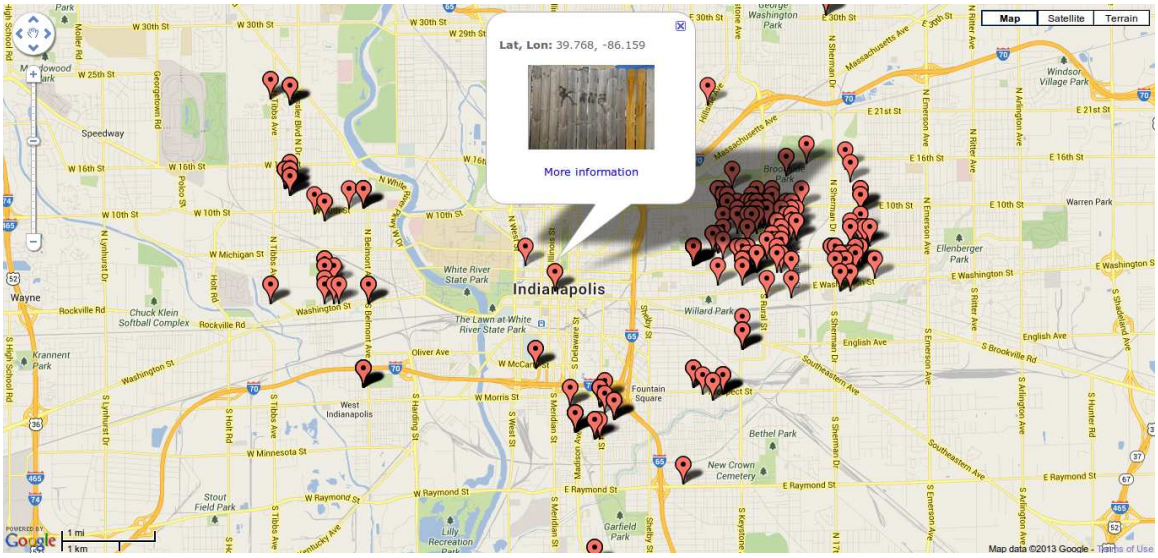


Fig. 3.83.: Example of a popped out balloon on the interactive map when a marker is clicked.

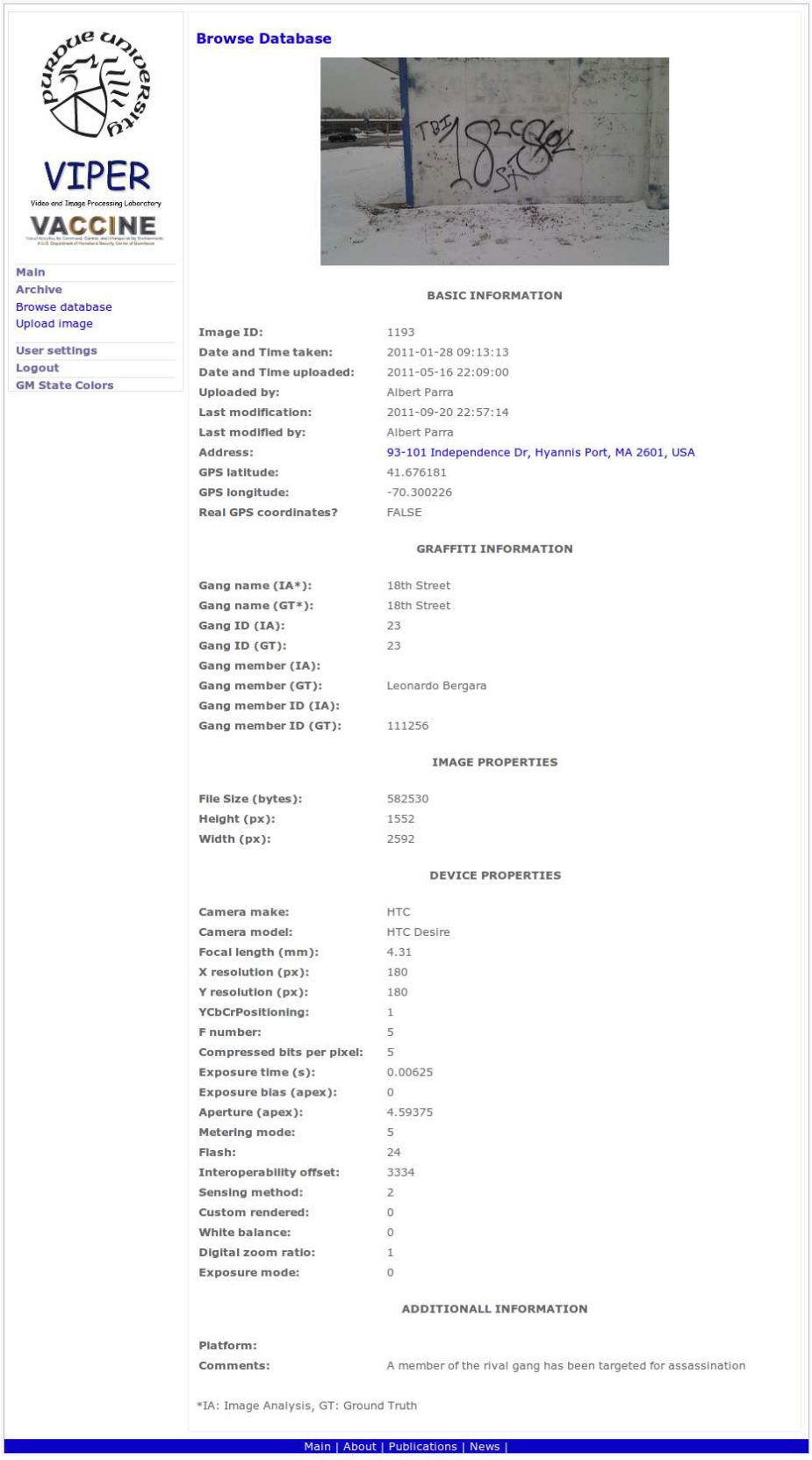


Fig. 3.84.: Example of "More information" result for a specific search in the database.

Fig. 3.85.: "Upload Image" Section of Desktop GARI.

Fig. 3.86.: Preview of an Image Before Uploading It to the Graffiti Database.

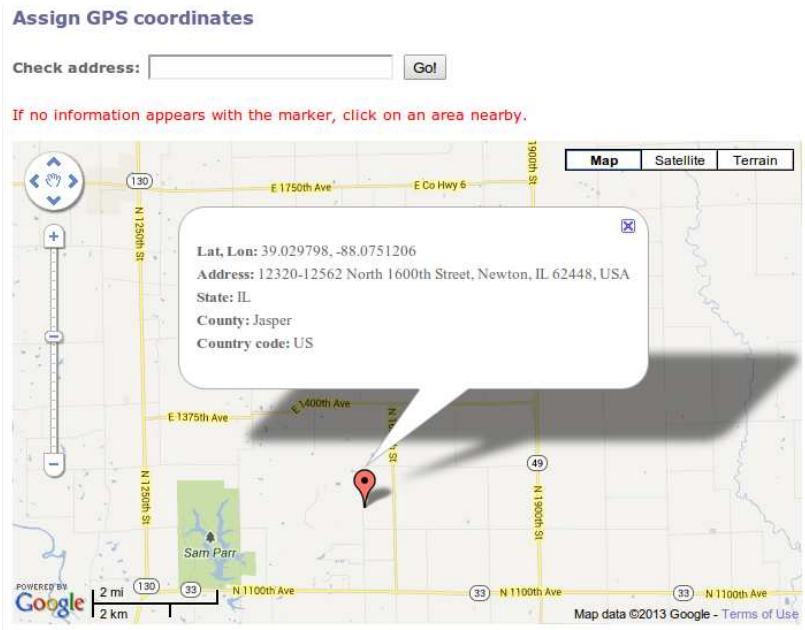


Fig. 3.87.: After uploading the image to the database, the user can select where the image was taken using an interactive map.

Fig. 3.88.: After uploading the image to the database, information can still be added.

Fig. 3.89.: Upload multiple images: Main screen.

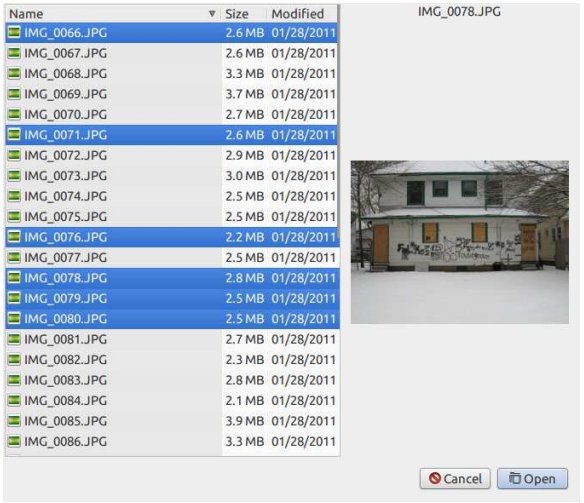


Fig. 3.90.: Upload multiple images: Select multiple files. Note that the appearance of this screen may vary depending on the operating system used.

Fig. 3.91.: Upload multiple images: List of images to upload.

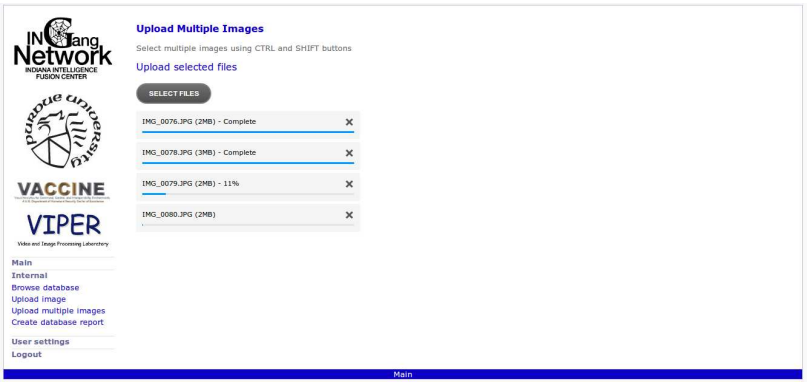


Fig. 3.92.: Upload multiple images: Upload progress.

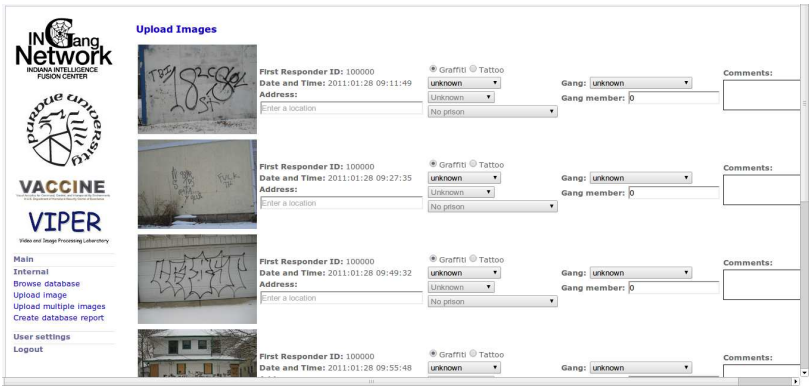


Fig. 3.93.: Upload multiple images: Review screen.

Create Database Report

The "Create database report" feature (Figure 3.94) allows a user to download a spreadsheet containing information from the database.

The available fields are:

- Image ID
- Path to the image file
- First responder name
- First responder ID
- Upload date and time
- Image size
- Image height
- Image width
- Camera make
- Camera model
- GPS longitude
- GPS latitude
- Address
- City
- County
- State
- ZIP code

- Country
- Comments

Multiple fields can be selected using the SHIFT or CTRL buttons on the keyboard. By holding SHIFT when clicking on two fields, it will select everything in between them. By holding CONTROL when clicking on fields, it will select individual fields. The number of entries to be downloaded range from 200 to all the entries on the database (i.e. all images on server). The entries to be downloaded can also be sorted by date and time. After clicking on "Submit" a spreadsheet is automatically created, and a link to the download is provided to the user, as shown in Figure 3.95.

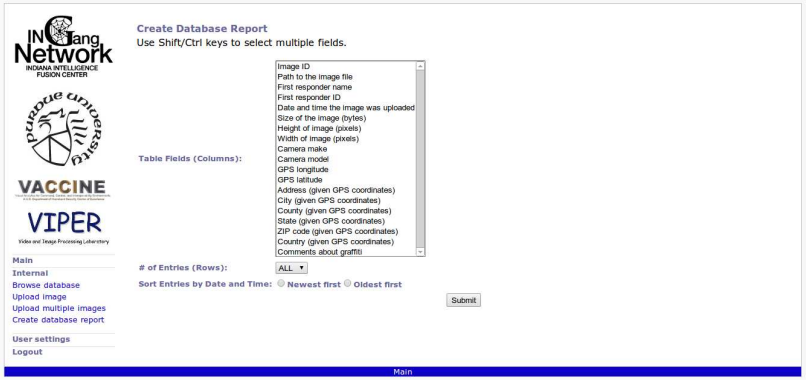


Fig. 3.94.: Create database report.

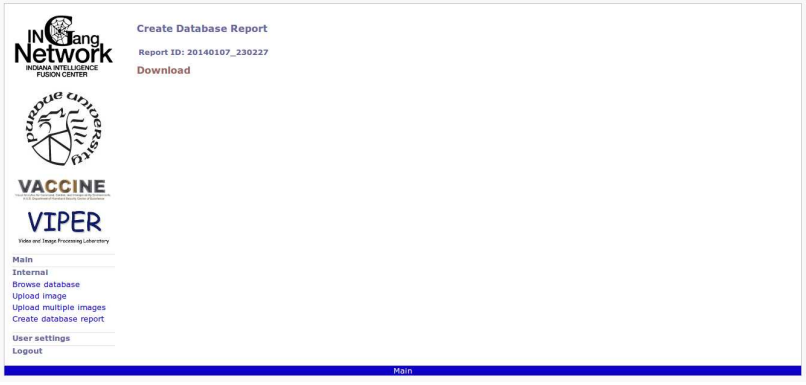


Fig. 3.95.: Create database report: download screen.

Security

Access and navigation to the web interface are established and managed using encrypted Secure Sockets Layer (SSL) sessions. SSL encrypts information both during the transmission. The user must log in using authorized credentials before entering the archive. Figure 3.96 shows the login page. Once successfully logged in an SSL session is created and maintained for the current user. The user account can be managed by clicking on the "User Settings" link on the left sidebar. Note that currently the only option available is password change.

Fig. 3.96.: Login Page for Accessing the Gang Graffiti Archive.

4. MOBILE EMERGENCY RESPONSE GUIDE (MERGE)

4.1 Review of Existing Methods

In this section we review some relevant literature in the areas of sign location detection and sign recognition.

4.1.1 Sign location detection

Sign location detection methods can be classified into three main categories: shape-based [248], color-based [249] and saliency-based [250].

Shape-based approaches first generate an edge map and then use shape information to find objects. For example, in [251] triangular, square and octagonal road signs are detected exploiting properties of symmetry and edge orientations exhibited by equiangular polygons. In [252] a road-sign detection system is based on support vector machines (SVM). It uses shape classification using linear and Gaussian-kernel SVMs. In most cases, the methods are invariant to translation, rotation, scale, and, in many situations, to partial occlusions. In [253] the authors present a system for detection and recognition of road signs with red boundaries and black symbols inside. Pictograms are extracted from the black regions and then matched against templates in a database. They propose a fuzzy shape detector and a recognition approach that uses template matching to recognize rotated and affine transformed road signs. In [254] the authors propose a system for automatic detection and recognition of traffic signs based on maximally stable extremal regions (MSERs) and a cascade of support vector machine (SVM) classifiers trained using histogram of oriented gradient (HOG) features. The MSER offers robustness to variations in lighting conditions. The system works on images taken from vehicles, operates under a range of weather

conditions, runs at an average speed of 20 frames per second, and recognizes all classes of ideogram-based (nontext) traffic symbols from an online road sign database.

Other shape-based approaches use "shape descriptors", which can be generally classified into two methods: contour-based methods and region-based methods [255, 256]. Contour-based methods only exploit the boundary information while region-based methods exploit all the pixels within a region. Contour-based methods are widely used in many applications because of their simplicity [168]. Although shape signatures obtained through contour-based methods are not generally robust to noise [168] the Fourier descriptor (FD) overcomes noise sensitivity by usually using only the first few low frequency coefficients to describe shape. The FD is also compact and easy to normalize. Because of its properties the FD is one of the most used shape descriptors [255–259]. In addition, it has been shown that the FD outperforms many other shape descriptors [168, 260].

Previous work on FDs includes methods for generating descriptors invariant to geometric transformations and matching methods for shape similarity and image retrieval. For example, in [261] a new Fourier descriptor is proposed for image retrieval by exploiting the benefits of both the wavelet and Fourier transforms. A complex wavelet transform is first used on the shape boundary, and then the Fourier transform of the wavelet coefficients at multiple scales is examined. Since FDs are used at multiple scales, the shape retrieval accuracy improves with respect to using ordinary FDs. FDs are analyzed as feature vectors in [262] for pedestrian shape representation and recognition. The results showed that only ten descriptors of both low and high frequency components of pedestrian and vehicle shapes are enough for accurate recognition. Shape context from [185] is used in [263] to generate descriptors and proposed a matching method that uses correspondences between two shapes based on ant colony optimization. In [264] the authors describe simple shapes using FDs based on chain codes and the Fourier transform. The first ten coefficients are used to approximate the shapes. In [257] the authors use the Fourier transform of local regions on the output of a MSER detector. They propose a FD matching method that

uses the phase information to extract the orientation of the shape and used the FDs for recognizing road signs. However, this method fails when signs have low resolution.

Color-based approaches overcome the problems of shape variation, partial occlusion, and perspective distortion. However, colors are sensitive to lightning conditions and illumination changes. To deal with these disadvantages, some color spaces that keep sign color almost invariant are used in existing methods. For example, in [265] sign detection is done using a color-based segmentation method as a preprocessing step for shape detection. Color-based segmentation is used to achieve real time execution, since color-based segmentation is faster than shape-based segmentation. In [266] several color components are used to segment traffic signs under different weather conditions. Various color spaces are analyzed to detect traffic prohibitive signs, alert signs and guide signs.

Saliency-based approaches utilize selective visual attention models, which imitate human early visual processing in order to overcome the above problems in complex scenes. This paper makes use of the saliency-based visual attention models to construct a hazmat sign saliency map as a sign localization method. Visual saliency is closely related to how we perceive and process visual stimuli and it is often characterized by variant object features, like color, contrast, gradient, edge, and contour. Theories of human visual attention hypothesize that the human vision system only processes parts of an image in detail while leaving others nearly unprocessed [267]. A saliency-based visual attention (SBVA) model was presented in [250] using images features with a Gaussian pyramid. A graph-based visual saliency (GBVS) method was proposed in [268], to highlight conspicuous regions. This method allows combinations with other visual attention maps. A dynamic visual attention (DVA) model based on the rarity of features is proposed in [269]. A histogram-based contract (HC) method and a region-based contract (RC) method were introduced in [270] to construct saliency maps. HC-maps produce better performance over RC-maps but at

the expense of increasing the computation time. A multi-scale dissimilarity aggregation (MSDA) method is used to estimate the saliency of regions in [271]. A saliency map generation method was described in [272] using image signature (IS) to highlight sparse salient regions based on RGB or Lab color spaces. An saliency detector based on hypercomplex Fourier transform (HFT) is presented in [273] using the convolution of the image amplitude spectrum with a low-pass Gaussian kernel.

4.1.2 Sign recognition

Sign recognition methods can be classified into: geometric constraint methods, boosted cascades of features, and statistical moments [274–276].

Methods based on geometric constraints include the use of Hough-like methods [277,278], contour fitting [279,280], or radial symmetry detectors [281,282]. These approaches apply constraints on the object to be detected, such as little or no affine transformations, uniform contours, or uniform lightning conditions. Although these conditions are usually met, they cannot be generalized. For example, [278] presents an analysis of Hough-like methods and confirms that the detection of signs under real-world conditions is still unstable. A novel Hough-like technique for detecting circular and triangular shapes is also proposed, in order to overcome some of the limitations exposed.

Methods based on the boosted cascades of features commonly use the Viola-Jones framework [283–285]. These approaches often use object detectors with Haar-like wavelets of different shapes, and produce better results when the feature set is large. For example, in [284] a system for detection, tracking, and classification of U.S. speed signs is presented. A classifier similar to the Viola-Jones detector is used to discard objects other than speed signs in a dataset of more than 100,000 images. In [285] the detection is based on a boosted detectors cascade, trained with a version of Ad-

aboost, which allows the use of large feature spaces. The system is robust to noise, affine deformation, partial occlusions, and reduced illumination.

Methods based on statistical moments [286–288] use the central moments of the projections of the object to be detected. They can be used to check the orientation of the object, or to distinguish between different shapes such as circles, squares, triangles, or octagons. These methods are not robust to projective distortions or non-uniform lightning conditions. For example, in [288] a mobile-based sign interpretation system uses detection of shapes with an approximate rotational symmetry, such as squares or equilateral triangles. It is based on comparing the magnitude of the coefficients of the Fourier series of the centralized moments of the Radon transform of the image after segmentation. The experimental results show that the method is not robust to projective distortions.

4.2 Segment Detection Using Geometric Constraints

Figure 4.1 shows the block diagram of the proposed method. We find edges in the image using the Canny edge detector. Since hazmat signs can be present at various distances, we use median auto-thresholding. To deal with non-uniform illumination changes in the scene, we also grayscale histogram equalize the image. We assume: 1) any sign in the image has to be approximately upright with its major axes aligned with the XY axis; and 2) the projective distortion has to be small. (i.e., edges have to be approximately at $\pm 90^\circ$ with respect to each other).

Given these assumptions, we use morphological filters to eliminate edges not belonging to a hazmat sign. We create flat linear structuring elements of length $L_{se} = 10$ pixels at $\theta_{se} = \pm 45^\circ$ and use them separately to erode the Canny edge map. Figure 4.2 shows the structuring elements used for erosion.

The resulting edge map is the superposition of the two erosions. We then find line segments using the Standard Hough Transform [226, 227] (already explained in

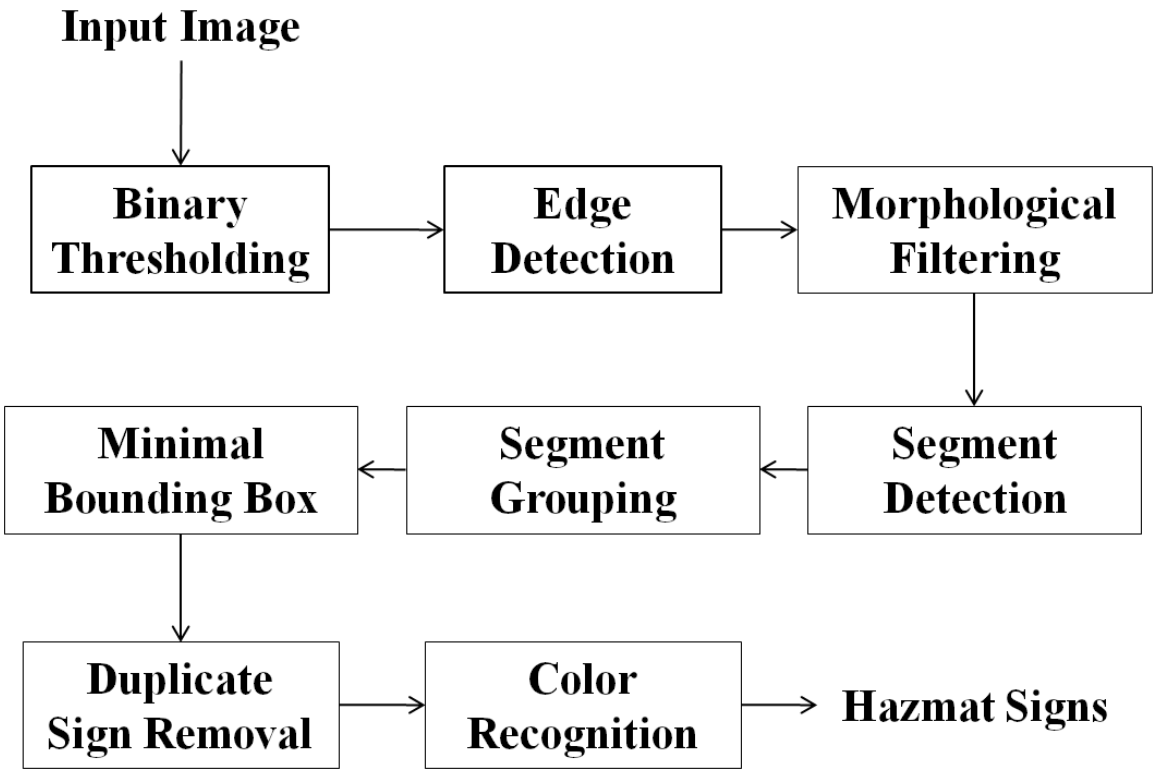


Fig. 4.1.: Segment Detection Using Geometric Constraints.

0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0

(a) Linear Structuring Element at $+45°$

1	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	1

(b) Linear Structuring Element at $-45°$

Fig. 4.2.: Structuring Elements Used for Erosion.

Section 3.5.3). We set the minimum gap allowed between points on the same line to $N_{gap}^{L} = 5$ pixels and the maximum gap to $N_{gap}^{H} = 0.05max(W_X, H_X)$, where (W_X, H_X) are the width and height of the image respectively.

We next proceed to group the segments into candidates. Each candidate consists of a set of segments having one reference segment, at least one parallel segment, and two orthogonal segments (one to the left and one to the right of the reference segment). The reference segment is chosen at random from the list of segments that have not been grouped yet. Parallel segments need to have similar slope and length relative to the reference segment. The thresholds are set so that $|m_p - m_r| < T_m$ and $|l_p - l_r| < T_l$, where m_p and m_r are the slopes of the parallel and reference segments respectively, l_p and l_r are the lengths of the parallel and reference segments respectively, $T_m = 0.1$, $T_l = 0.75e$ and $e = max(l_p, l_r)$. The distance d between the reference and the parallel segments has to be in the range $T_d^L < d < T_d^H$, where $T_d^L = 0.5e$ and $T_d^H = 2.5e$. This distance is defined between the middle points of the parallel and the reference segments. Also, the angle between the reference and the parallel segments has to be less than $\theta_{RP} = 20°$. This angle is defined by the normal of the parallel segment at its middle point and the vector joining the middle points of the parallel and the reference segments. Orthogonal segments need to have opposite slope and similar length to the reference segment, that is, $|m_o + 1/m_r| < T_m$ and $|l_o - l_r| < T_l$, where m_o and l_o are the slope and the length of the orthogonal segment. The distance d between the reference and the orthogonal segments has to be in the range $T_d^L < d < T_d^H$. The angle between the reference and the orthogonal segments is defined as positive when the orthogonal segment is to the right of the reference segment, and defined as negative when the orthogonal segment is to the left of the reference segment.

For each candidate set satisfying the geometric constraints we compute its minimal bounding box. We then discard any candidate with a bounding box aspect ratio smaller than $T_{BB} = 1.3$.

Finally, we check the remaining candidates and remove those that correspond to the same sign. This can be done by first dividing all bounding boxes that overlap more than $T_{overlap} = 50\%$ into groups, and then finding the optimal bounding box for each group. We consider the optimal bounding box to be the one with its nodes closest to its centroid (i.e. closest to a square).

Figure 4.3 illustrates an example of the complete process. Once a hazmat sign is segmented, its color is set to the average hue inside the optimal bounding box and the color is used to identify the sign. We also do basic text recognition inside the detected region using the open source Optical Character Recognition (OCR) engine OCRAD [289]. Although the accuracy of OCRAD is far below other state-of-the-art OCR engines, it was chosen for its speed [290]. Note that the text recognition step is applied just for testing purposes. Other text recognition methods will be investigated in the future (see Section 6).

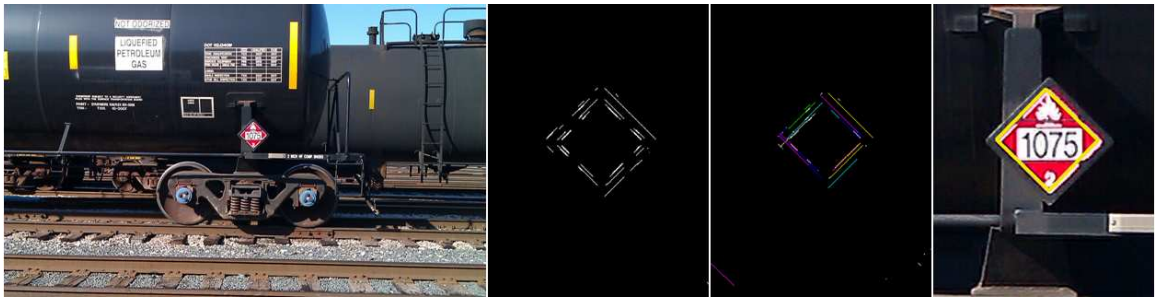


Fig. 4.3.: First method (left to right): original image, segments at $\pm 45°$, grouped segments, optimal bounding box.

Table 4.1 shows all the parameters/thresholds we used including empirically derived parameters.

4.3 Convex Quadrilateral Detection Based on Saliency Map[1]

Our first method described above has some drawbacks:

[1]The work presented in this section was done by the author jointly with Bin Zhao.

Table 4.1: Parameters and thresholds used in Segment Detection Using Geometric Constraints. W_X and H_X are the width and height of X respectively. $e = max(l_p, l_r)$

Parameter	**Description**	**Value**
L_{se}	Length of structuring elements for erosion	10 px
θ_{se}	Orientation of structuring elements for erosion	$\pm 45°$
N_{gap}^{L}	Maximum gap for Standard Hough Transform	5 px
N_{gap}^{H}	Minimum gap for Standard Hough Transform	$0.05max(W_X, H_X)$
T_m	Slope threshold	0.1
T_l	Length threshold	$0.75e$
T_d^L	Low distance threshold between segments	$0.5e$
T_d^H	High distance threshold between segments	$2.5e$
θ_{RP}	Angular threshold between segments	$20°$
T_{BB}	Bounding box ratio threshold	1.3
$T_{overlap}$	Bounding box overlap threshold	50%

- **Grayscale**: By converting the original RGB image to grayscale we loose color information. This can cause the hazmat sign to have similar intensity values as the background given specific illumination conditions. Figure 4.4 illustrates an example. The edge detection process cannot separate the top corner of the sign from the background, thus loosing the necessary edges to continue the recognition process.

- **Low resolution/Blurry**: With low resolution or blurry images, the resulting edge map will not contain straight edges at $\pm 45°$ and the erosion process will then delete most of them. Figure 4.5 shows an example.

- **Distortion**: Hazmat signs not satisfying the two assumptions of the first method will be removed during the erosion process. Figure 4.6 shows an example.

- **Line overlap**: The gap threshold of the Standard Hough Transform may cause the segment grouping process to merge two segments from two close signs, as shown in Figure 4.7.

- **Shade**: The image contains shade that can alter the color of the sign. Figure 4.8 illustrates an example. The result is an unsuccessful color recognition once the hazmat sign is detected.

Our second technique replaces the initial edge detection with a saliency map to detect regions potentially containing hazmat signs[2]. The block diagram in Figure 4.9 shows the block diagram of the proposed method. Figures 4.10 and 4.11 illustrate examples of the saliency maps obtained on the Lab and RGB color spaces. Note how the saliency map applied on the RGB color space does better on black or white signs (low chroma region), while the Lab color space does better on the rest of the signs.

We apply visual saliency models to the input images represented in both RGB and Lab color spaces. In each color space, two saliency maps are constructed using

[2]This work was done by Bin Zhao.

Fig. 4.4.: Issue With First Method: Grayscale. Sign Is Lost On Line Detection Process.

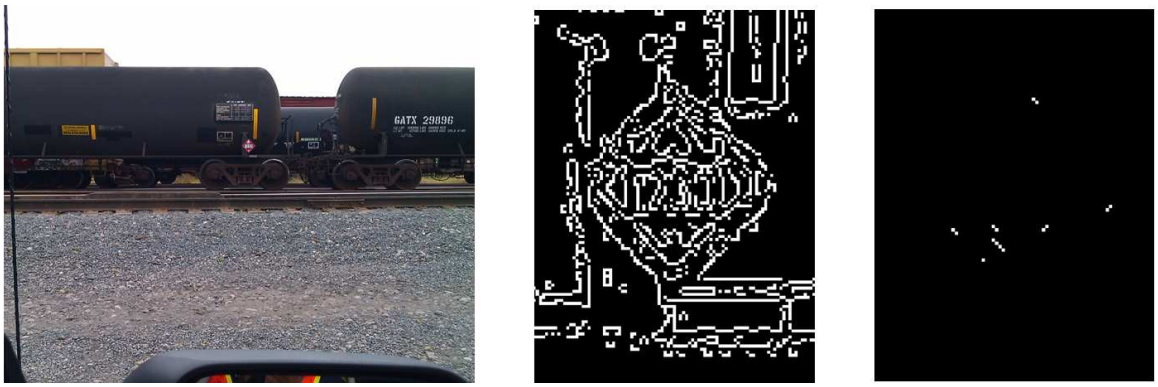


Fig. 4.5.: Issue With First Method: Low Resolution. Sign Is Lost On Erosion Process.

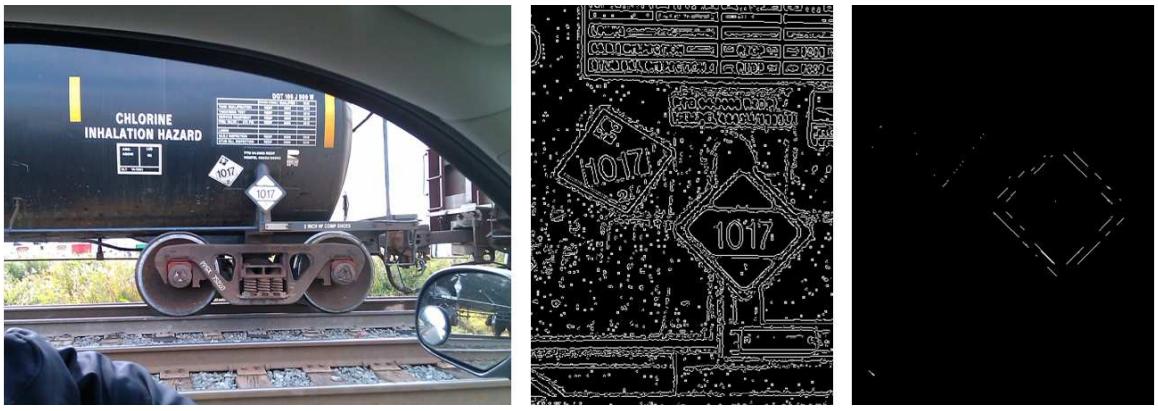


Fig. 4.6.: Issue With First Method: Sign Distortion. Sign Is Lost On Erosion Process.

Fig. 4.7.: Issue With First Method: Segment Merging. Sign Is Lost On Segment Grouping Process.

Fig. 4.8.: Issue With First Method: Shade. Sign Color Is Not Recognized Properly.

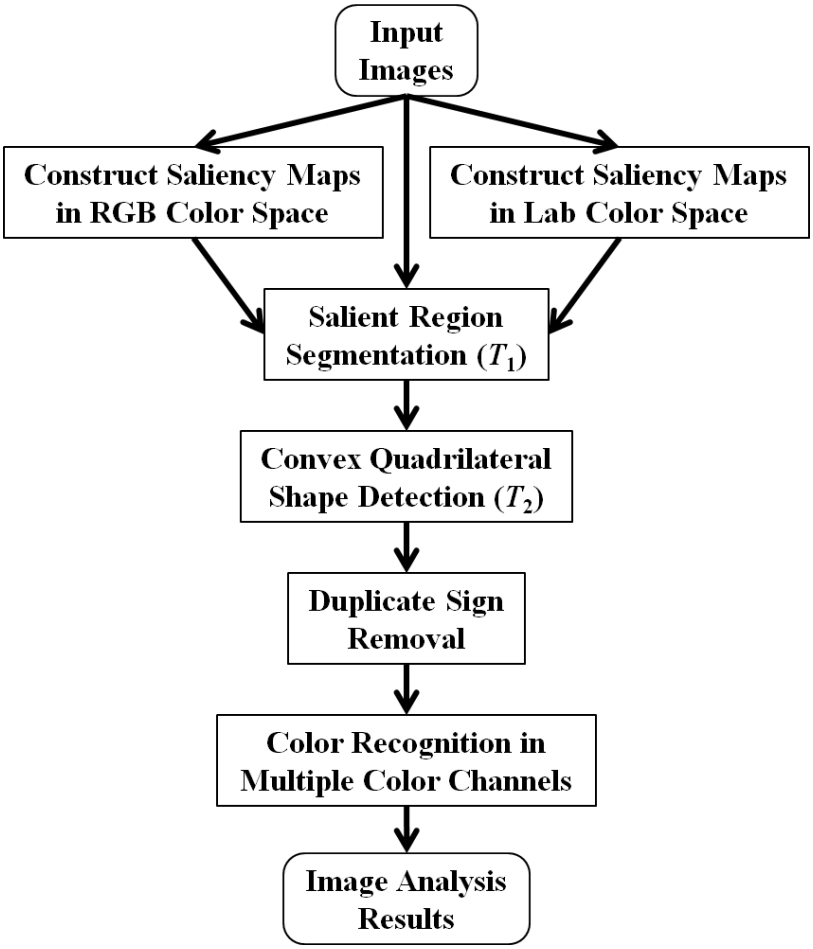


Fig. 4.9.: Proposed Hazmat Sign Detection and Recognition Method.

Fig. 4.10.: Saliency Map Method Obtained On Lab (Middle) and RGB (Right) Color Spaces.

Fig. 4.11.: Saliency Map Method Obtained On Lab (Middle) and RGB (Right) Color Spaces.

two visual saliency models separately, i.e. IS [291] and HFT [273]. The saliency maps assign higher saliency value to more visually attractive regions. Note that the original HFT method uses the I-RG-BY opponent color space. We modified this method to use RGB and Lab color components with different weights ($W_{RGB} = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$ for RGB and $W_{Lab} = [\frac{1}{2}, \frac{1}{4}, \frac{1}{4}]$ for Lab). The combined saliency map method, denoted as IS+HFT(RGB+Lab), generates four saliency maps (two for RGB and two for Lab) and produces the best results in the experiments (see Section 5.2.2). We threshold each saliency map to create a binary mask to segment the salient regions from the original image. The threshold T_1 is determined as k times the average saliency value of a given saliency map. That is, $T_1 = \frac{k}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} S(x, y)$, where W and H are the width and height of the saliency map, $S(x, y)$ is the saliency value at position (x, y) and k is empirically determined for the combined saliency map method ($k = 4.5$ for IS and $k = 3.5$ for HFT).

For each salient region found, we detect signs using specific color channels. Hazmat signs in our datasets contain either one or two of the following colors: black, white, red, blue, green, yellow. We then divide the input image into six color channels and we process them as separate images. The red, green and blue channels are obtained from the RGB color space. The yellow channel is obtained from the CMYK color space. The black and white channels are obtained by thresholding the Y channel.

This allows us to do both sign detection and color recognition at the same time, since we will assume that the color of any hazmat sign found in the region will correspond to the color channel associated to it. Note that although our dataset does not contain orange hazmat signs, they exist and can appear in the future. We would then be able to extract a seventh channel by transforming the image from RGB to a hue-based color space and then segment the hue channel.

The grayscale and the color channels are thresholded to account for highly chromatic areas using an empirically determined threshold T_2 (85 for black, 170 for white, and 127 for color). Note that this last threshold can be avoided by working with a hue-based color space. Each of the thresholded images is binarized, and morpholog-

ically opened to remove small objects containing less than $N_{px}^{O} = 0.05\%WH$. We also use dilation with a flat, disk-shaped structuring element of size $S_{se} = 7$ to merge areas that may belong to the same object. Figure 4.12 shows the structuring element used for dilation.

0	0	1	1	1	0	0
0	1	1	1	1	1	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
0	1	1	1	1	1	0
0	0	1	1	1	0	0

Fig. 4.12.: Structuring Element Used for Dilation.

We then retrieve the contours from the resulting binary image [292]. For each contour, we use the Standard Hough Transform [226, 227] to find straight lines that approximate the contour as a polygon. The intersections of these lines give us the corners of the polygon, which can be used to discard non-quadrilateral shapes. If the contour is approximated by four vertices, we find its convex hull [293]. If the convex hull still has four vertices, we check the angles formed by the intersection of its points. If each of these angles is in the range $T_{\theta}^{v} = 90^{\circ} \pm 1.5^{\circ}$, and the ratio of the sides formed by the convex hull is in the range $T_{r}^{e} = 1 \pm 0.5$, we can assume that we have found a convex quadrilateral.

Finally, we use the same technique as in the first method to remove quadrilaterals that correspond to the same hazmat sign. Figure 4.13 illustrates a successful detection of two signs, one is affected by rotation and perspective distortion. Figure 4.14 illustrates a successful detection of one sign and also a false positive. In this particular

case the issue could be addressed by using an optical character recognition to detect the text inside the sign candidate.

Fig. 4.13.: Second Method: True Positives.

Fig. 4.14.: Second Method: True Positive/False Positive.

Our second method offers multiple advantages. First, it is robust to rotation, since there is no erosion at $\pm 45°$. Second, it is robust to perspective distortion, since convex quadrilaterals can be skewed. Third, it is able to detect signs close to each other, since there is no overlapping of line segments caused by the Standard Hough Transform. Fourth, it is more robust to blurred and low resolution images, since there is no edge detection is on the sign recognition step. Lastly, it is more robust

Table 4.2: Parameters and thresholds used in Convex Quadrilateral Detection Based on Saliency Map. W and H are the width and height of the saliency map. $S(x, y)$ is the saliency value at (x, y)

Parameter	**Description**	**Value**
W_{RGB}	RGB weights for saliency model	$[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$
W_{Lab}	Lab weights for saliency model	$[\frac{1}{2}, \frac{1}{4}, \frac{1}{4}]$
T_1	Saliency map threshold	$\frac{k}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} S(x, y)$
k	Weight included in T_1 (IS)	4.5
k	Weight included in T_1 (HFT)	3.5
T_2	Color channel threshold (black)	85
T_2	Color channel threshold (white)	170
T_2	Color channel threshold (color)	127
N_{px}^{O}	Number of pixels for opening	$0.05\%WH$
S_{se}	Size of structuring elements for dilation	7
T_{θ}^{v}	Angular threshold between convex hull vertices	$90° \pm 1.5°$
T_r	Ratio threshold between convex hull edges	1 ± 0.5

to color recognition, since it detects signs already in specific color channels. The only disadvantage is its execution time. The first method uses basic geometry to find potential candidates, while the second method needs to compute a saliency map as a preprocessing step, which takes more time than the first process itself.

Table 4.2 shows all the parameters/thresholds we used including empirically derived parameters.

4.4 Sign Location Detection Based on Fourier Descriptors[3]

The second method is robust to geometric distortions and illumination changes. However, it relies on the detection of straight edges and the relationship between their lengths and angles. This causes the process fails on low resolution images, signs with partial occlusions and deteriorated signs. We propose a third method to overcome the drawbacks caused by detections based on geometric constraints. Figure 4.15 shows the block diagram of the proposed method. We use contour shape representation and

[3]The work presented in this section was done by Kharittha Thongkor jointly with the author and Bin Zhao.

matching based on Fourier descriptors. Note that we do not use a saliency map to get an initial sign location estimation. Instead we use the original image as input to our system.

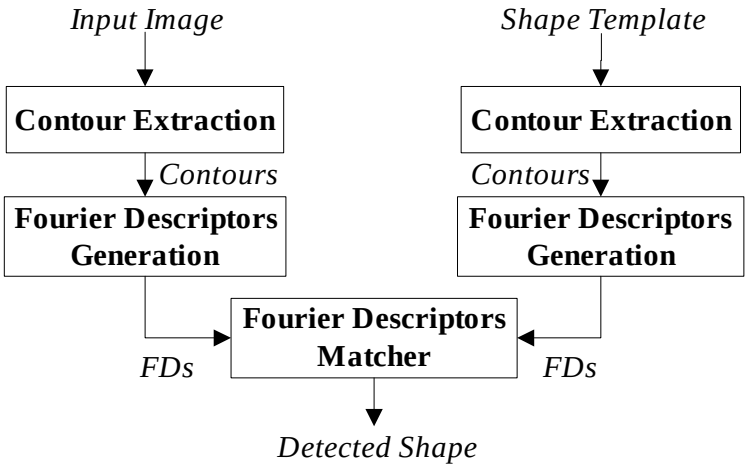


Fig. 4.15.: Sign Location Detection Based on Fourier Descriptors.

In this method we also detect hazmat sign locations in specific color channels, so no further color recognition is not required after detecting the location of the sign. As opposed to our second method, each of the six images extracted from each color channel is binarized separately. For this purpose we propose the use of color channel thresholding followed by Otsu's thresholding technique [294] to obtain the final binary image. For each of the six color channel images, I_i, $i \in [1, 6]$, we first select two parameters for channel thresholding, T_{i_1} and T_{i_2}. The reason why we need T_{i_1} and T_{i_2} is that directly using Otsu's thresholding method on a channel does not produce accurate results when images contain variable illumination [295]. Histogram of each color channel can be analyzed for minima/valleys which can then be used to determine two thresholds as follows. T_{i_1} is set to

$$T_{i_1} = min\Big(\frac{255}{4}, h_{i_1}\Big), \tag{4.1}$$

where H_{i_1} is the location of the first valley of the histogram of the i^{th} color channel. The first valley is the minimum point between the first two significant peaks. The

set of significant peaks P_1 of a histogram h is defined as the set of points with a histogram value greater than their local maximum neighbors [296]. That is,

$$P_1 = \{(p_i, h(p_i))|h(p_i) > \{h(p_{i-1}), h(p_{i+1})\}, p_i \in P_0\}, \tag{4.2}$$

where

$$P_0 = \{(i, h(i))|h(i) > \{h(i-1), h(i+1)\}, 0 \leq i \leq 255\}, \tag{4.3}$$

T_{i_2} is set to

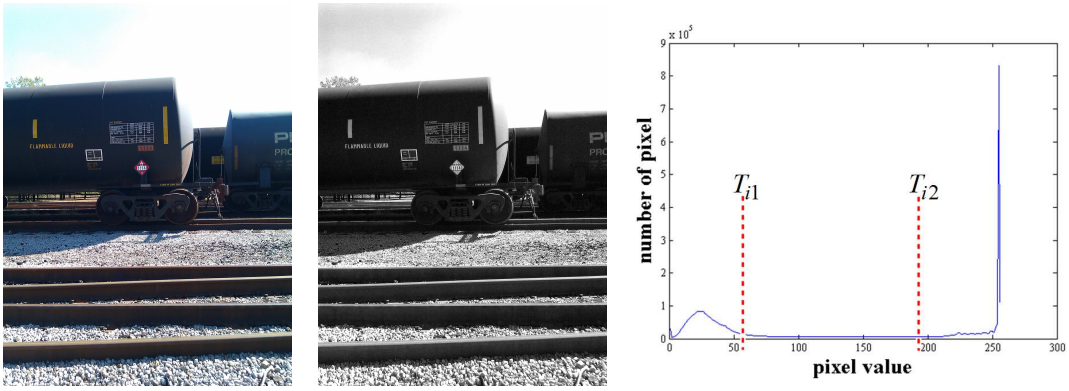
$$T_{i_2} = max\Big(3\frac{255}{4}, H_{i_2}\Big), \tag{4.4}$$

where H_{i_2} is the location of the last valley of the histogram of the i^{th} color channel. The color channel image I_i is then thresholded by:

$$I_i'(x,y) = \begin{cases} 0 & I_i(x,y) \leq T_{i_1} \text{ or } I_i(x,y) \geq T_{i_2} \\ I_i(x,y) & \text{otherwise} \end{cases} \tag{4.5}$$

Each image I_i' is then used as input for Otsu's thresholding method to automatically generate a threshold T_{i_b}. Finally, each original color channel image I_i is then binarized using T_{i_b}. Figure 4.16 illustrates a comparison using Otsu's method with and without our proposed color channel thresholding method. Note how Otsu's method fails to find the optimal threshold because of the high density of pixels in the sky region having high intensity values in the red channel.

As we mentioned above we use morphological techniques to merge areas in the binary image found above that may belong to the same hazmat sign. First, we use a flood-fill operation to fill holes in the binary image [297]. A hole is a set of background pixels surrounded by foreground pixels. Next, we use morphological dilation with a flat, diamond shape structuring element of size $S_d = 5$ pixels to enlarge the boundaries of foreground areas [256, 298]. Then, we remove small objects by using morphological opening with a flat, diamond-shaped structuring element of size $S_o = 20$ pixels. We

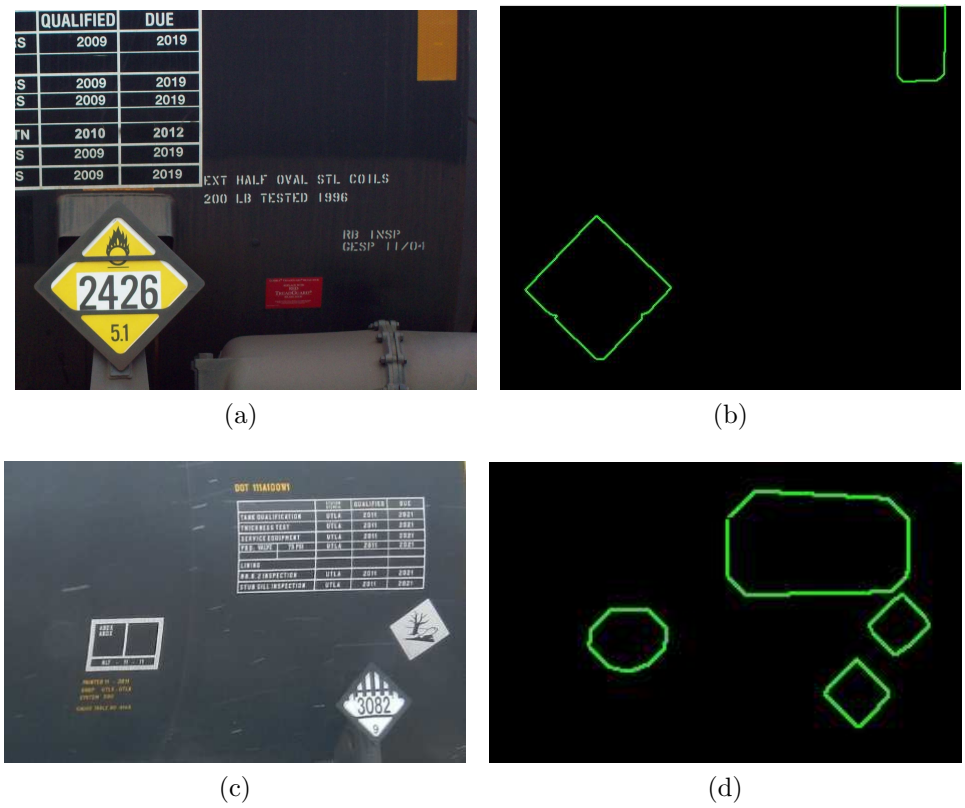


(a) Original image (b) Red channel (c) Red channel histogram

(d) Otsu's method (e) Proposed method

Fig. 4.16.: Example of image binarization using our proposed color channel thresholding method comparing with Ostu's method.

also remove objects containing less than $T_c = 0.03\%$ of the total number of pixels in the image. We chose 0.03% because it is the minimum number of pixels contained in a hazmat sign in our image test set. Finally, we obtain closed contours by tracing the exterior boundaries of objects in the resulting binary image [299, 300]. Figure 4.17 shows some examples of extracted contours from input images. Note that the size of the structuring elements are empirically obtained from the ground-truth data in our dataset. They came from searching the best values that give the maximum number of signs before tracing the exterior boundaries of objects.



(a) (b)

(c) (d)

Fig. 4.17.: Examples of input images (left) and their contours (right).

Each contour found from the previous step is used to generate a Fourier Descriptor (FD). The FD describes the shape of an object through the use of the Fourier transform of the object's contour. Assuming the contour of a shape has N pixels,

numbered from 0 to $N-1$, a set of coordinates describing the contour can be defined as

$$b(k) = (x(k), y(k)) = x(k) + iy(k), \tag{4.6}$$

where $k = 0, 1, 2, \ldots, N-1$. The Fourier transform of the contour function, $A(v)$, is the FD:

$$A(v) = F(b(k)) = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} b(k) exp^{-\frac{j2\pi vk}{N}}, \tag{4.7}$$

where $v = 0, \ldots, N-1$. To describe the shape of a boundary the Fourier coefficients have to be normalized to make them invariant to translation and scale [169, 257, 261, 262, 264].

If the 2D shape is translated by a distance $z_0 = x_0 + jy_0$:

$$b'(k) = b(k) + z_0 \tag{4.8}$$

its FD becomes

$$A'(v) = \frac{1}{N} \sum_{k=0}^{N-1} (b(k) + z_0) exp^{-\frac{j2\pi vk}{N}} \tag{4.9}$$

$$= \frac{1}{N} \sum_{k=0}^{N-1} b(k) exp^{-\frac{j2\pi vk}{N}} + \frac{1}{N} \sum_{k=0}^{N-1} z_0 exp^{-\frac{j2\pi vk}{N}} \tag{4.10}$$

$$= A(v) + z_0 \delta(v). \tag{4.11}$$

This means the translation only affects the DC component $A(0)$ of the FD. Therefore, by setting the first coefficient, $A(0)$, to zero we make the FD invariant to translation.

If the 2D shape is scaled (with respect to origin) by a factor S:

$$b'(k) = Sb(k) \tag{4.12}$$

its FD is scaled by the same factor:

$$A'(v) = SA(v). \tag{4.13}$$

Therefore, by normalizing the energy of the remaining coefficients to 1 we make the FD invariant to scale. The normalized FD $A'(v)$ then becomes:

$$A'(v) = \frac{A(v)}{\sqrt{\sum_{v=1}^{\infty} |A(v)|^2}},\ A'(0) = 0. \tag{4.14}$$

The low frequency components of $A'(v)$ contain information about the general shape and the high frequency components contain finer details. Therefore, the first P Fourier descriptor coefficients can be used to create an approximate reconstruction of the contour $b(k)$,

$$\widehat{b}(k) = \frac{1}{P} \sum_{v=0}^{P-1} A'(v) exp^{\frac{j2\pi vk}{N}}, k = 0, 1, 2, \ldots, N-1. \tag{4.15}$$

In order to determine if a contour obtained from an image belongs to a hazmat sign we need to compare its FD against the FD of a predefined shape template or shape contour in a process called contour matching. In this paper the shape template is a diamond shaped binary image resembling a hazmat sign (see Figure 4.18). Contour matching can be done in the spatial or frequency domain. We use matching in the frequency domain for two reasons. First, matching in the frequency domain is scale independent, as opposed to spatial domain matching. Second, matching in the spatial domain involves scanning an image multiple times modifying the scale and rotation of the shape template. Since the normalized FDs are invariant to scale and the correlation matching in frequency domain is invariant to rotation the matching is less computationally expensive. The frequency domain matching has also been shown to be more efficient [301, 302] and allows easy recognition for rotated and scaled noisy sign images [170].

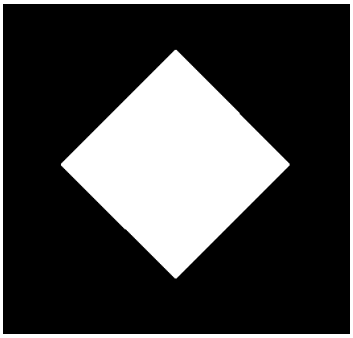


Fig. 4.18.: A diamond shaped binary image is used as a shape template.

FD matching is usually done by using only the magnitude and ignoring the phase information. By discarding the phase information we achieve rotation and starting point invariance [303]. This is because a rotation of the 2D shape by an angle ϕ about the origin only introduces a phase shift in the FD:

$$b'(k) = b(k)e^{j\phi} \Rightarrow A'(v) = A(v)e^{j\phi}, \tag{4.16}$$

and a shift of the 2D shape from 0 to m_0 only introduces a phase shift in the FD:

$$b'(k) = b(k - m_0) \Rightarrow A'(v) = A(v)e^{\frac{j2\pi m_0 v}{N}}. \tag{4.17}$$

However, different shapes can have similar magnitude but completely different phase information, thus making magnitude-based matching less accurate [257]. Therefore, we use a correlation-based matching cost function that uses both magnitude and phase information [257]. The cross-correlation between the shape template contour T and the image contour I, $r_{TI}(l)$ is

$$\begin{aligned} r_{TI}(l) = (T * I)(l) &= \int_0^K \overline{\mathrm{T(k)}}\mathrm{I(l + k)}\,\mathrm{d}k \\ &= \sum_{v=0}^{\infty} \overline{A'_T(v)} A'_I(v) exp^{-\frac{j2\pi vl}{K}} \\ &= F^{-1}\{\overline{A'_T} A'_I\}(v). \end{aligned} \tag{4.18}$$

$A'_T(v)$ and $A'_I(v)$ are the normalized FDs of the template and the input contours, respectively.

By using normalized contours and complex FD matching we approximately compensate for scaling, rotation, translation and starting point. We say "approximately" because we are only using the first few Fourier coefficients to describe the shape of the contour. To find the appropriate number of Fourier coefficients needed for matching we examined the effect of varying the number of low-frequency coefficients we used

from our shape template. Figure 4.19 illustrates the effect of using the first 2, 5, 8, 16, 30, 50, 80 and 100 coefficients from our shape template. Using more Fourier coefficients than necessary leads to increasing computation time with no additional benefit. Adding too many coefficients does not significantly improve the matching performance [168]. Thus, only the first eight Fourier coefficients were used in our experiments.

To decide if a contour extracted from an image corresponds to a hazmat sign we need some way of matching the normalized FD of our shape template and the normalized FD of the extracted contour. Correlation-based matching estimates the cost between two normalized FDs. The cost is defined as

$$e = 2 - 2 \max_{l} |r_{TI}(l)|, \tag{4.19}$$

where $|\cdot|$ denotes the complex modulus. Thus we check if the correlation-based matching cost e between the normalized FD of our shape template and the normalized FD of the extracted contour is below a threshold T_e. To obtain the value of T_e we calculate the correlation-based matching cost e between our shape contour (Figure 4.18) and each of the shape template contours shown in Figure 4.20. Since the cost of matching our shape template against a diamond shape (including rotation) is not greater than 1.75 we set $T_e = 1.75$. Note that the shape templates in Figure 4.20 are only used to decide the value of T_e.

Table 4.3 shows all the parameters/thresholds we used including empirically derived parameters.

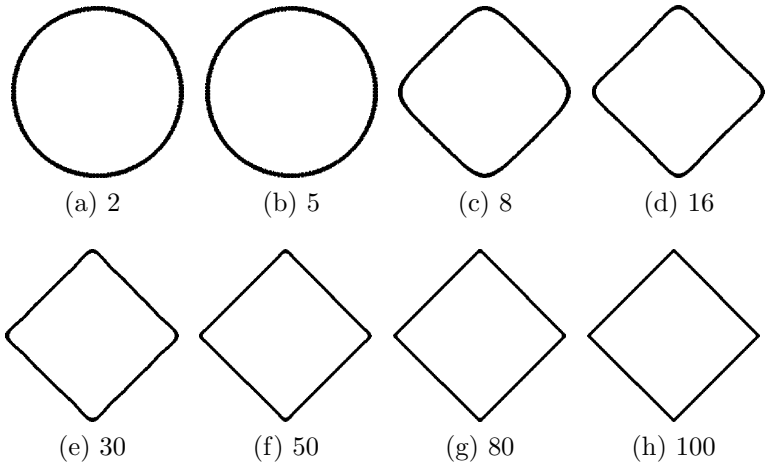


Fig. 4.19.: Reconstruction of the shape template using the first 2, 5, 8, 16, 30, 50, 80 and 100 Fourier coefficients.

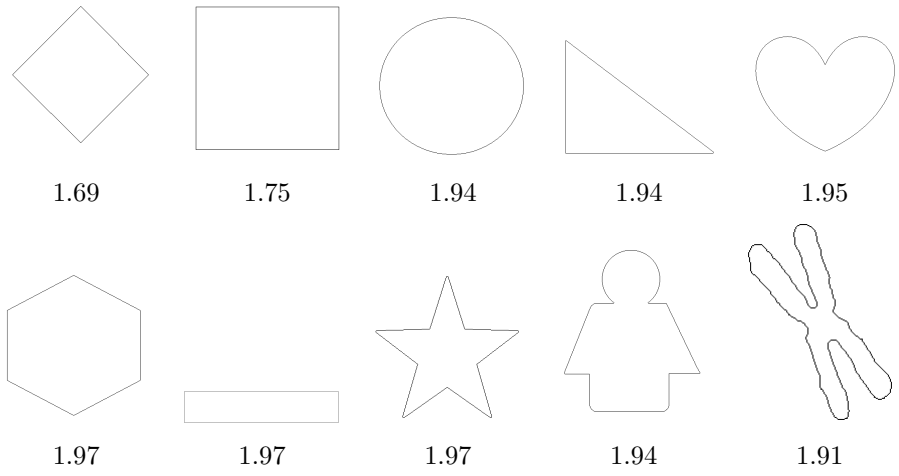


Fig. 4.20.: Comparison of our shape template contour against different shape templates and their matching costs e.

Table 4.3: Parameters and thresholds used in our proposed method. Automatically determined values are denoted by *. W and H are the width and height of the image.

Parameter	**Description**	**Value**
T_{i_1}	Low threshold for channel thresholding	*
T_{i_2}	High threshold for channel thresholding	*
T_{i_b}	Otsu's threshold for binarization	*
S_d	Size of structuring element for dilation	5 px
S_o	Size of structuring element for opening	20 px
T_c	Connected components threshold	$0.03WH$
T_e	Correlation-based matching cost threshold	1.75

4.5 System Implementation[4]

4.5.1 System Overview

We implemented a prototype of the MERGE system as an application for Android and iOS devices and as a web-based interface accessible from any web browser. Figure 4.21 illustrates the MERGE system, which is divided in two groups:

1. **Client-side**: Browse an internal database on the Android device, consisting of the contents of the ERG 2012 Guidebook[5]. Figure 4.22 illustrates the client-side system.

2. **Server-side**: Use image analysis on the server and communicate the results back to the client. Figure 4.23 illustrates the server-side system.

The client-side includes the device and methods available to the users, operating without the use of a network connection. The offline services are only available from Android devices (Section 4.5.3). The online services are available from both Android devices or any web browser (e.g., Internet Explorer, Mozilla Firefox, Google Chrome). This includes desktop and laptop computers as well as Blackberry smartphones (Section 4.5.4). The server-side includes the image analysis process to detect and interpret the hazmat signs.

[4]The work presented in this section was done by the author jointly with Andrew W. Haddad.

[5]The internal database was initially created by Andrew W. Haddad and later updated by the author.

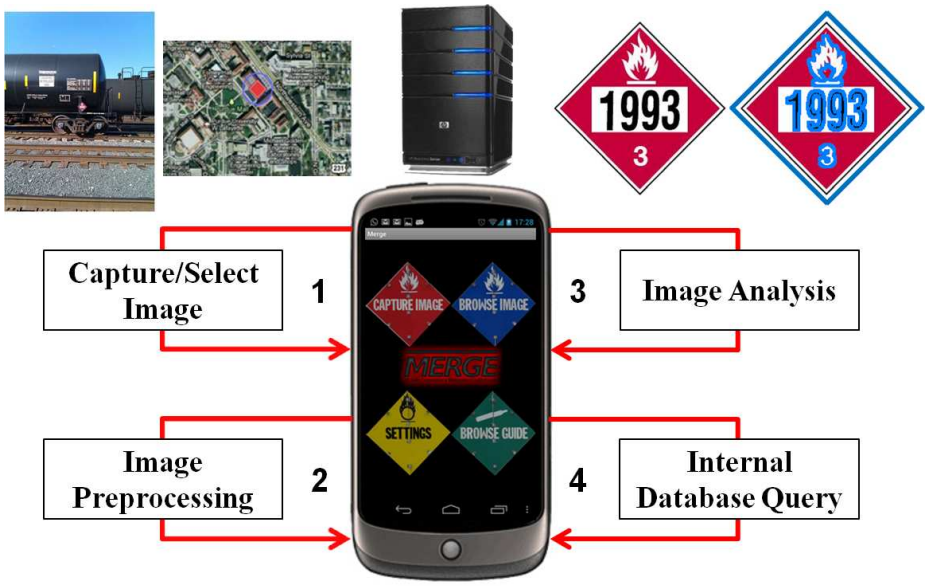


Fig. 4.21.: Mobile-Based Hazmat Sign Detection and Recognition.

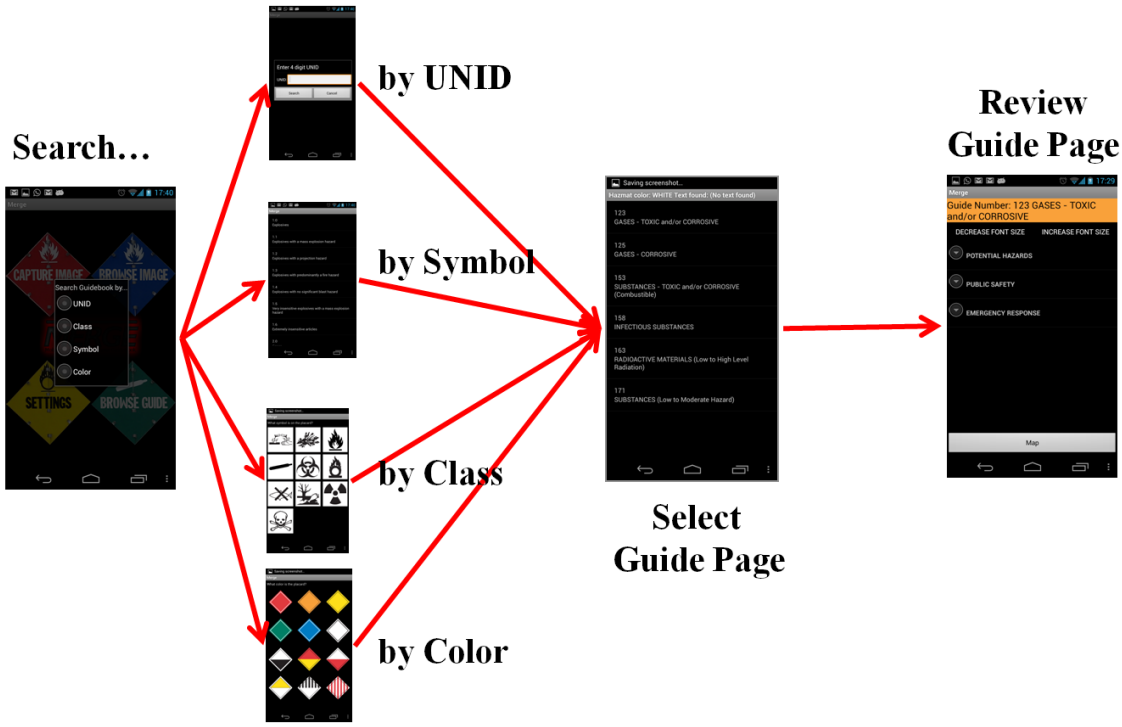


Fig. 4.22.: Overview of the MERGE Client-Side Components.

4.5.2 MERGE Databases

In this section we describe how the image database is organized. We will first describe the database schema and then show by an example how the information GARI acquires is added to the database. The database of hazmat signs was deployed for three reasons:

1. To collect and organize images acquired by first responders. This includes images of hazmat signs, images of scenes for forensic analysis, and metadata.

2. To store the results of the image analysis.

3. To manage first responders' credentials, allowing them to access the services available through the Android/iOS applications and the web based interface.

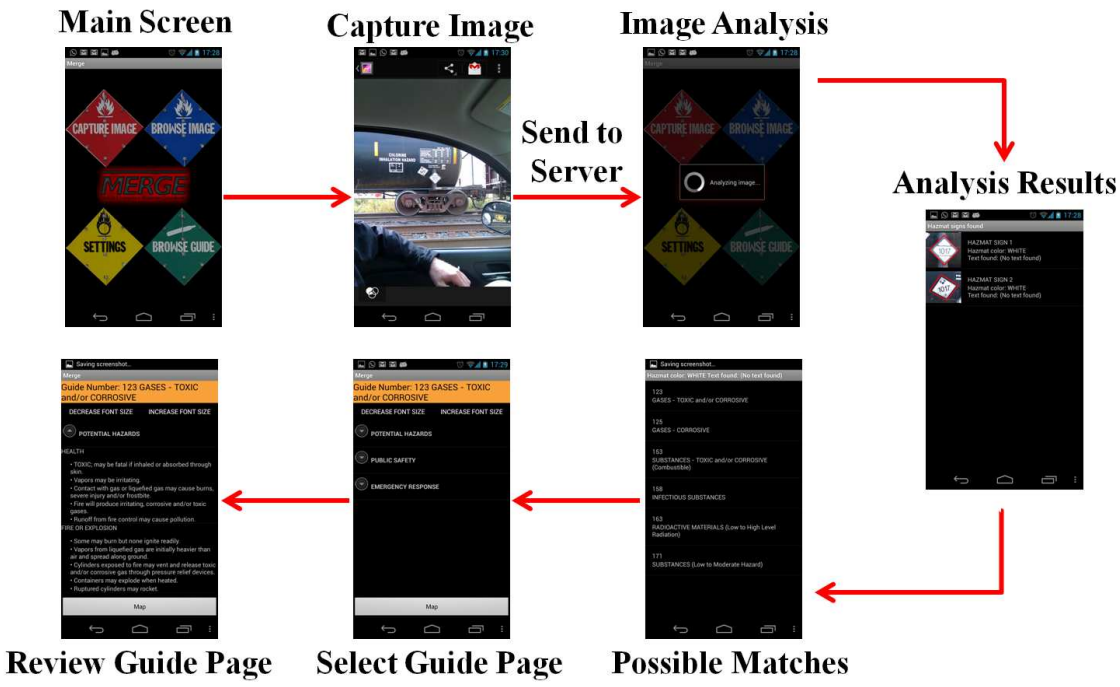


Fig. 4.23.: Overview of the MERGE Server-Side Components.

The MERGE database is implemented in PostgreSQL on a Linux server. It consists of 15 tables, all of them detailed in Appendix E. Figure 3.48 illustrates the structure of the 11 main tables. Note that the schema does not show all the fields in all the tables but just the relevant fields to indicate the association between the tables.

1. **images**: Stores EXIF data from the images along with image location and general image information and the results from the image analysis. The fields related to this table are shown in Tables E.1, E.2 and E.3 in Appendix E.

2. **vw_01_orange_page**: Stores the relationships between guide page numbers, guide pages, categories and details.

3. **vw_03_yellow_page**: Stores the relationships between guide page numbers and UNIDs.

4. **vw_05_water_reactive_materials**: Stores relationships between UNIDs, dangerous goods and guide page numbers.

5. **placard**: Stores the relationships between UNIDs, placards, symbols and classes.

6. **unids**: Stores the relationships between guide pages, UNIDs and hazardous materials.

7. **class**: Stores information about classes.

8. **colorPages**: Stores the relationships between guide pages and placard colors.

9. **textPages**: Stores information about the text contained in the guide pages.

10. **symbols**: Stores information about the symbols that can appear in hazmat signs.

11. **users**: Stores users' credentials to access to the system services as well as information concerning administrative privileges, email addresses, and registration and login status. Table E.4 in Appendix E describes the fields of this table.

Note that currently we only populate the tables **images** and **users**.

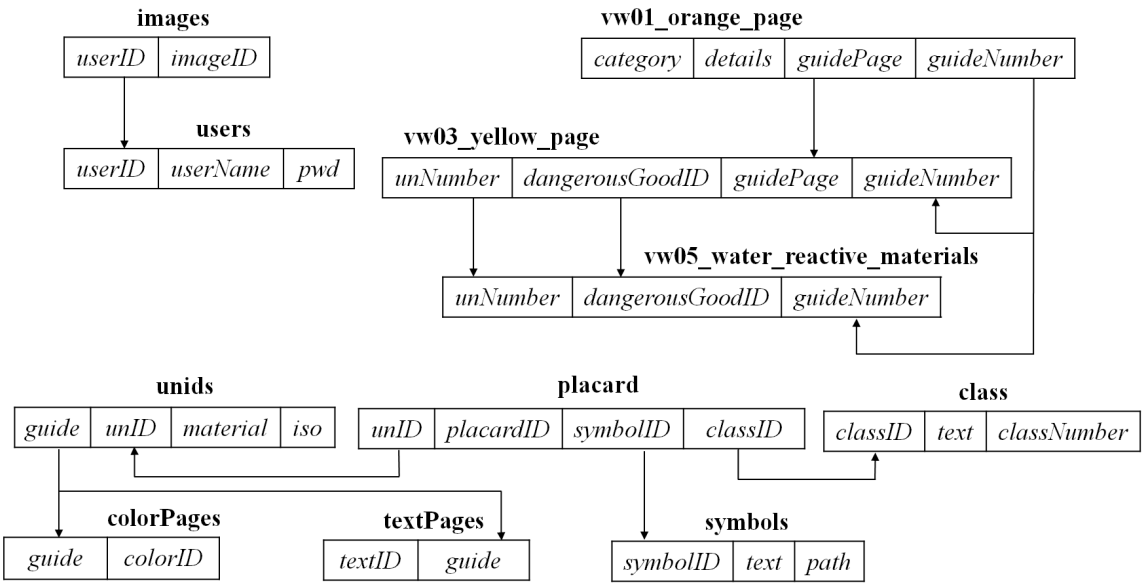


Fig. 4.24.: Database Schema Showing The Associations Between the Tables in the Database.

4.5.3 Android/iOS Implementation

We implemented the MERGE system on Android and iOS devices as summarized in Figures 4.22 and 4.23. We called this application Mobile MERGE. In this section we describe how the application works and describe its user interface.

Overview

A user takes an image of the scene containing one or multiple hazmat signs using the embedded camera on the device via the Graphical User Interface (GUI). The EXIF data of the image, including GPS location and date and time of capture, is automatically added to the image header. The image is then automatically sent to the server for analysis. The results are sent back to the user, and links to an internal database are provided. The internal database is a digitized version of the 2012 ERG.

Another option is to browse the internal database. The user can search for information about hazmat signs by UNID, symbol, class, or color. Each of the options provide links to the guide pages containing information to determine what specialty equipment, procedures and precautions should be taken in the event of an emergency.

We implemented the system on different smartphones makes and models, but always targeting version 3.0 of the Android operating system (OS).

User Interface[6]

Our Android application does not require the use of a network connection. However it is mandatory if the user wants to update the application or analyze an image. The application automatically checks for updates when launched, notifying the user if a new version is available (Figure 4.25). A user must be assigned a User ID and a unique password in order to use MERGE. Once the User ID and password has been entered, the main screen is shown (Figure 4.26). The main screen includes the following options, which are described below:

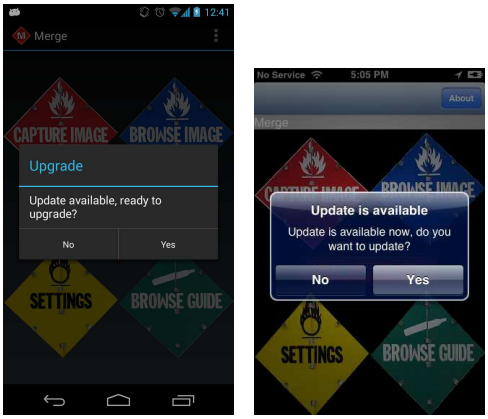
- Browse Image
- Browse Guide Pages
- Capture Image
- Settings
- About

Note that the "About" option appears when the user presses the menu button.

Browse Image

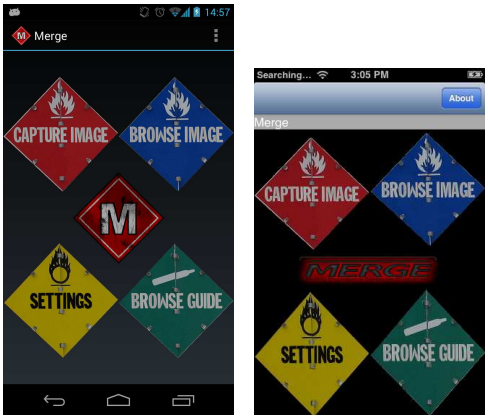
The user has the option to browse images stored on the Android device, instead of taking an image using the "Capture Image" option. Note that the entire phone image gallery is browsed, including images that have not been taken using the Mobile

[6]The user interface was initially created by Andrew W. Haddad, and later updated by the author.



(a) Android (b) iPhone

Fig. 4.25.: Automatic updates.



(a) Android (b) iPhone

Fig. 4.26.: Main Screen.

MERGE application. When the option “Browse Image” is tapped, a directory browsing window is opened, and the user can search and select the desired image. Figure 4.27 shows an example of browsing. Once the image is selected, it is automatically sent to the server for analysis.

(a) Android (b) iPhone

Fig. 4.27.: Screens for browsing images.

Browse Guide Pages

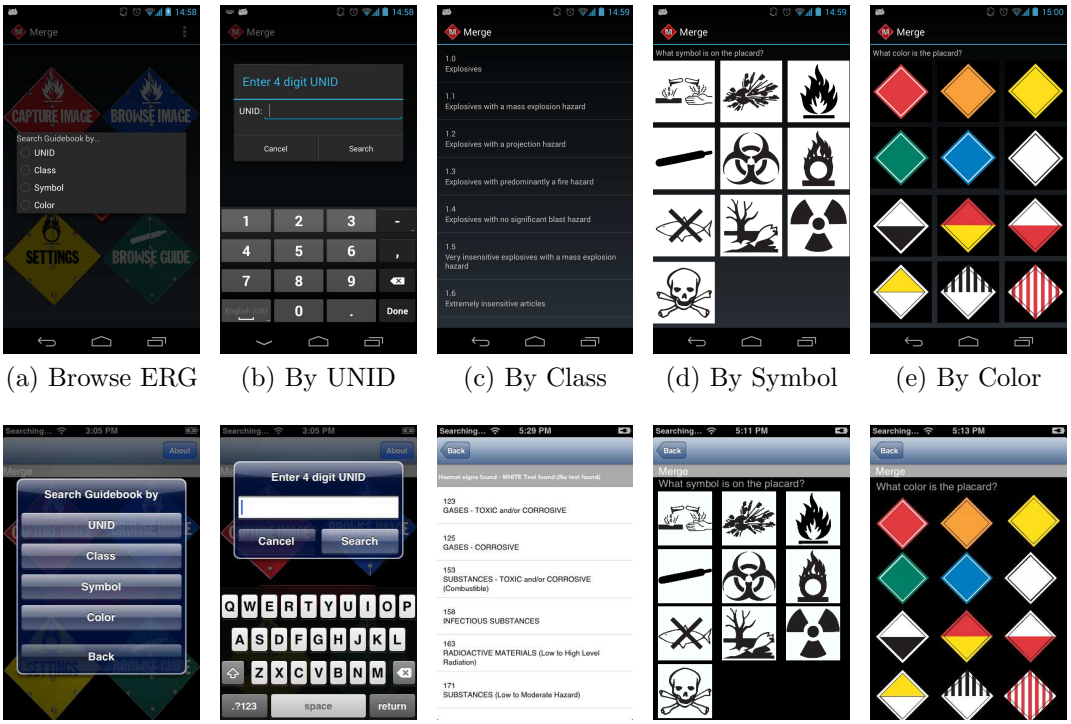
When the user chooses to search for a guide page, they are presented with a dialog containing four different ways to search the database, depending on what information is available to the user. The four options are shown in Figure 4.28:

1. UNID

   The four-digit UNID number should be one of the UNIDs found in the 2012 emergency response guidebook. The valid range for guide pages is 1001-9279. Numbers outside this range will produce an error indicating the proper range.

2. Class

   Each class produces a list of pages or a single guide page pertaining to the particular class selected. In many cases, the list cannot be narrowed automatically and the decision is left to the user. The possible classes are: Explosives, Gases, Flammable Liquids, Flammable Solids, Oxidizing Substances, Toxic Substances, Corrosive Substances, and Miscellaneous Hazardous Materials.



(a) Browse ERG (b) By UNID (c) By Class (d) By Symbol (e) By Color

(f) Browse ERG (g) By UNID (h) By Class (i) By Symbol (j) By Color

Fig. 4.28.: Methods for browsing. Android (top) and iPhone (bottom).

3. Symbol

   Though symbols are often related to the guide pages similarly to the classes, they do not always match. Often we have multiple symbols per class and multiple classes per symbol. Similarly to classes, each symbol produces a list of pages or a single guide page pertaining to the particular symbol selected. In many cases, the list cannot be narrowed automatically and the decision is left to the user. The possible symbols are: Corrosive, Explosive, Flammable, Gases, Infectious, Oxidizing, Pollutant, Radioactive, and Toxic.

4. Color

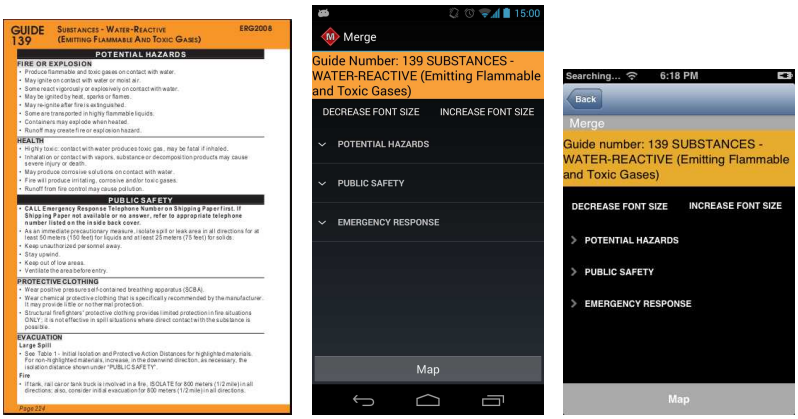
   Each color or combination of colors represents a number of guide pages. Similarly to classes and symbols, each color produces a list of pages or a single guide page pertaining to the particular color or combination of colors selected. In many cases, the list cannot be narrowed automatically and the decision is left to the user. The possible colors and combinations of colors are: Red, Orange, Yellow, Green, Blue, White, White and Black, Red and Yellow, White and Red, Yellow and White, White and Black Stripes, White and Red Stripes.

5. Guide Page

   The ERG contains a section where the general hazards of the dangerous goods are covered (orange-bordered pages, also known as guides). Each guide is divided into three main sections: potential hazards, public safety, and emergency response (Figure 4.29a). The guides in Mobile MERGE are organized in the same fashion as in the ERG, but using expandable lists. That is, the user can search for a specific guide page and tap on any of the three sections to read all the information available (Figure 4.29c).

   (a) Page Number

   The first thing the user will notice, at the top most of the orange header, is the Guide Page number. This is made available so the user can cross-



(a) 2012 ERG (b) Android MERGE (c) iPhone MERGE

Fig. 4.29.: Guide page in the ERG 2012 and corresponding guide page in Mobile MERGE for Android (middle) and iPhone (right).

reference the information provided by MERGE with the Emergency Response Guidebook (ERG) 2012 if necessary.

(b) Substance

Next, also in the header, the user will see the substance name/category. E.g. Oxidizers.

(c) Categories

As previously stated, the page is separated into categories, subcategories, and details. The headers for possible categories are: Potential Hazards, Public Safety, Emergency Response, Supplemental Information.

(d) Map

If a green table entry is available for a given guide page, the user will be presented with the option of displaying a map with a recommended evacuation region defined according to the current location of the user and the chemical chosen. Figure 4.30) shows the steps followed to obtain the evacuation region. The user will be asked up to three questions to better define the evacuation region: "Large of Small Spill?", "Initial Isolation or Protective Action?" and "Is it Day or Night?". After the user answers these questions, a map is displayed. The map will always contain a circle shape indicating the evacuation region, and for some available chemicals it will also contain a plume model, as seen in Figure 4.30h. The plume shape is obtained by querying database of real-time weather information, which provides more accurate evacuation information using wind speed and direction at the current location.

Capture Image

If the user taps the "Capture Image" button from the main screen an image can be acquired. The camera interface, shown in Figure 4.31, allows the user to take an image of a hazmat sign to be analyzed ("SIGN") or an image of the scene for future forensic analysis ("SCENE").

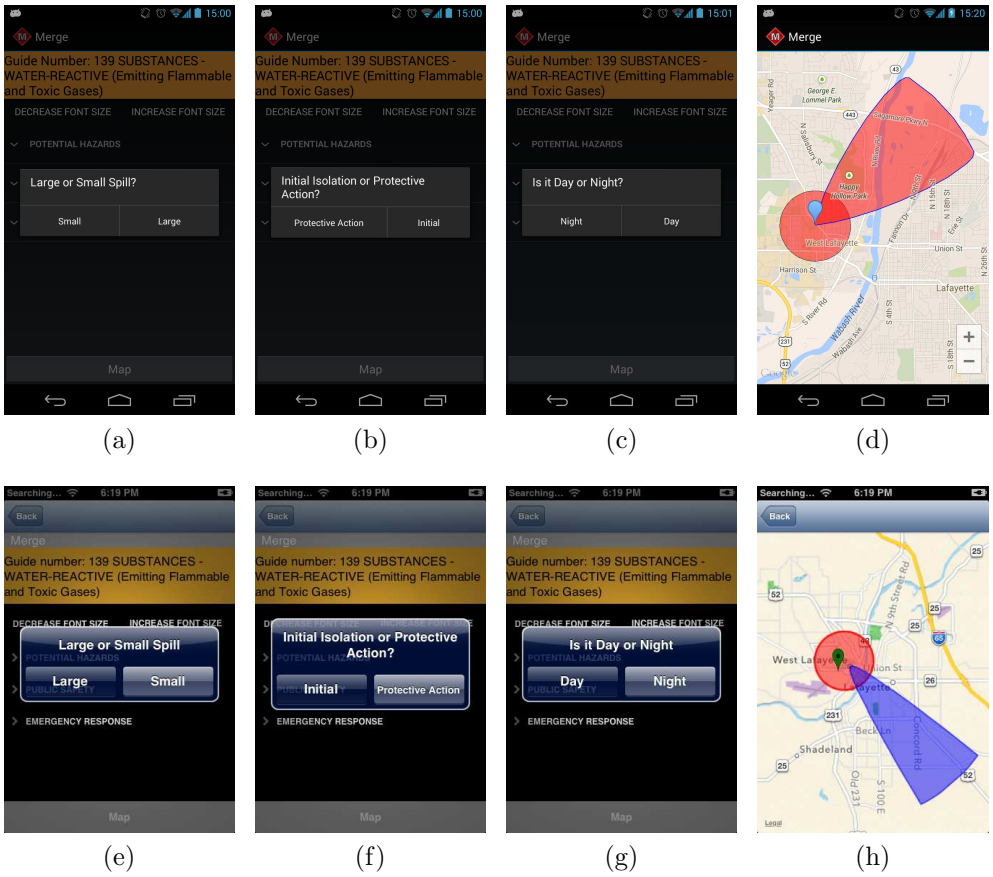


Fig. 4.30.: Evacuation region for Android (top) and iPhone (bottom). From left to right, questions asked to refine evacuation region, and general evacuation circle and weather-based plume model.

The image is automatically sent to the server, where it is be stored and analyzed. The user is notified through a dialog that the image upload and analysis is taking place. The analysis is done only when the image is captured using the "SIGN" option. After the analysis, the user will be presented with options to determine correctness and the closest matching guide page associated with the captured placard.

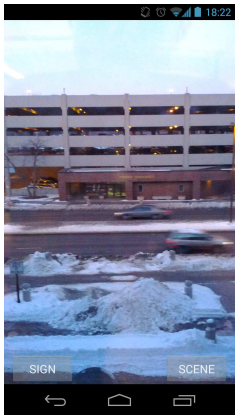


Fig. 4.31.: Camera Interface with "SIGN" and "SCENE" options.

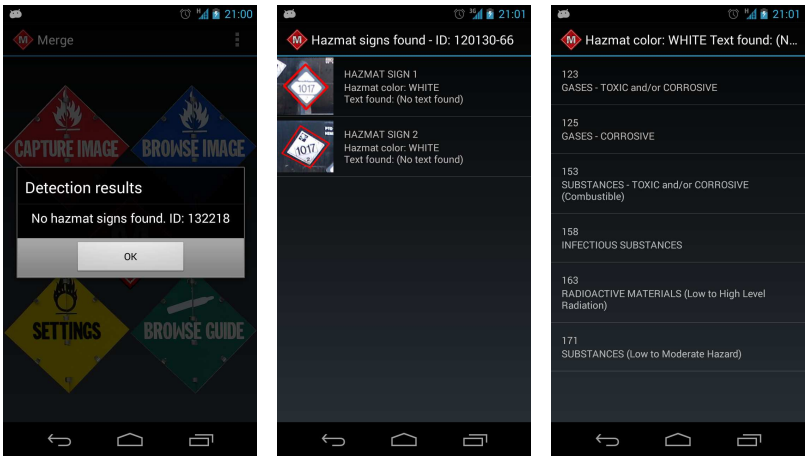
When the image analysis is completed, the results are shown to the user. There are two possible scenarios.

1. If no placard has been found a dialog informs the user (Figure 4.32d).

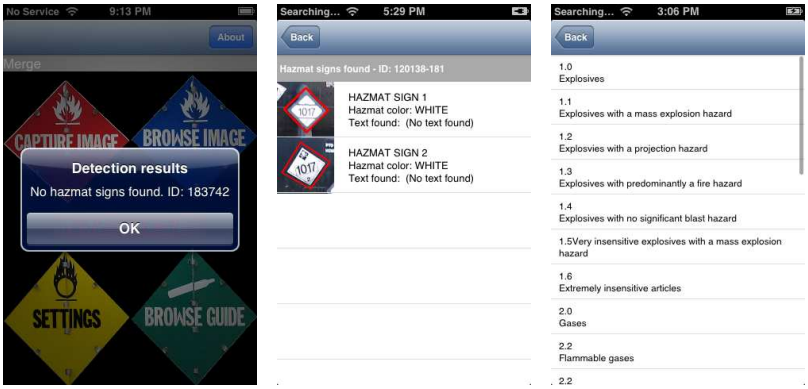
2. If the system has successfully determined which placard the image corresponds to, it will show a list with the results (Figure 4.32e). If more than one result is possible for a particular placard (e.g., if the placard color is found but not the text) a list of all the associated guide pages are shown (Figure 4.32f); otherwise, a single guide page is shown (Figure 4.29c).

Security

Our Android application is used by first responders from multiple agencies. Therefore, it is mandatory to ensure that only authorized users can access and use the



(a) No Placard Found (b) Results of Analysis (c) Possible Guide Pages



(d) No Placard Found (e) Results of Analysis (f) Possible Guide Pages

Fig. 4.32.: Results of the Image Analysis Process. Android (top) and iPhone (bottom)

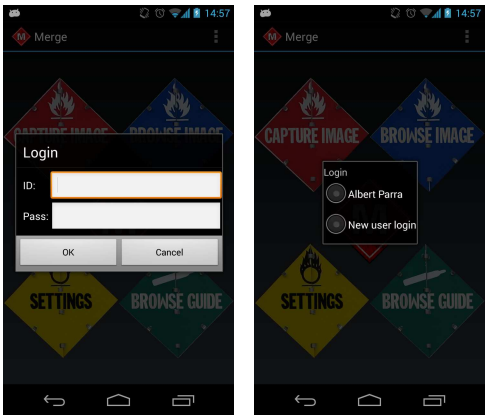
application. The connections to the server must be secure and all the information transmitted to and from the server must be encrypted (using the SSL/TLS protocol). The user credentials are sent every time the application contacts the server to make sure the connection is made by an authorized user. In the Android version we use ProGuard [246], a code optimizer and obfuscator for the Android SDK. It reduces the application size up to 70% and makes the source code more difficult to reverse engineer. It also improves the battery life by removing verbose logging code in a background service. An additional level of security includes the creation of two types of users:

- Regular users: Can switch between users, change their password, delete specific images only taken by themselves, and send crashlogs to the server.
- Administrative users: Can modify the server domain name/IP address, change user IDs, change passwords, delete specific images from any user, delete all images of any specific user, and send crashlogs to the server.

When launching the MERGE application, a dialog box prompts the user (Figure 4.33a). The user ID and a password is entered. If this is the first time the user logs in, a new dialog box prompts the user to change the default password (provided by the MERGE staff by email). For successive logins, the user will appear on a list of previously logged users, and no password is necessary (Figure 4.33b).

All authorized users can access the "Settings" option from the main screen of the application. Figure 3.73 shows the various options.

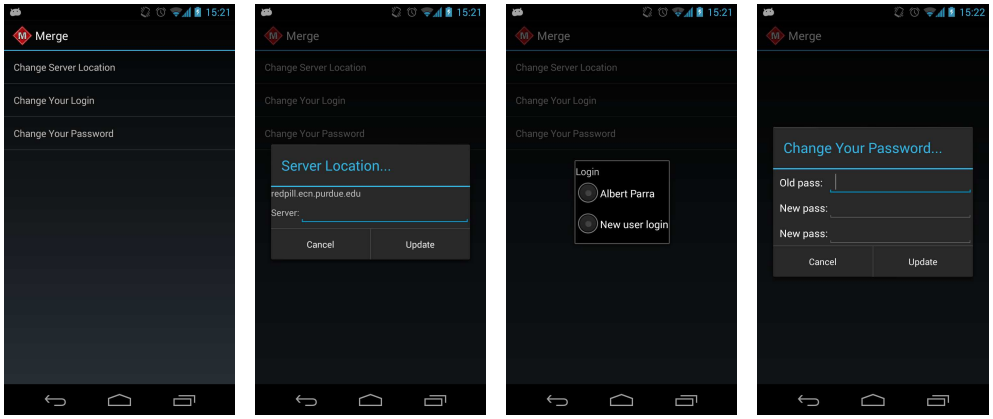
- Server Location: Administrative users can change the server IP address (Figure 4.34f).
- Change Login: The change user dialog is exactly the same as the login dialog, with the exception that if someone chooses to change the user for the application, they can cancel the change before submitting (Figure 4.34g).



(a) First time login (b) Returning user

Fig. 4.33.: User ID Screen.

- Change Password: The change password dialog is exactly the same as the change password dialog, which appears immediately after the first login - on either the website or the mobile app - with the exception that if someone chooses to change the password for the current user of the application, they can cancel the change before submitting (Figure 4.34h).



(a) Settings Menu (b) Change Server (c) Change User (d) Change Password

(e) Settings Menu (f) Change Server (g) Change User (h) Change Password

Fig. 4.34.: Settings Menu Options. Android (top) and iPhone (bottom).

4.5.4 Web Interface[7]

System Overview

We also implemented our system as a web interface tthat gives the user access to the hazmat database, and provides the ability to upload and browse images, and browse the official guidebook. We called this application Desktop MERGE. The web interface is available from any device with a web browser. This includes all desktop and laptop machines and all mobile telephones capable of browsing the web (e.g., iPhone, Blackberry, Android devices).

User Interface

As of March 2014 the MERGE website is located at www.hazmat-signs.org. The main page contains information about the MERGE project, its principal investigators, and the graduate students involved. The "Internal" page (Figure 4.35) displays the options the user has to interact with the graffiti database, including Browse Guidebook and Browse Images.

Browse Guidebook

Users can browse the guidebook using four different methods (Figure 4.36). The intersection of the sets created by the Color, Symbol and Classes chosen will be returned as a list of guide pages. Given more information, users can combine colors with symbols and classes. This will produce a smaller list of placards, containing all of the characteristics added. When a user searches by UNID, it takes preference over the other fields. That is, if UNID is searched, the Color, Symbol and Class fields are ignored.

The list of results contains images representing the color and symbol and shows the class searched. Each entry in the list of results contains the Guide Page number and Guide Page name (Figure 4.37).

[7]The work presented in this section was done by Andrew W. Haddad.

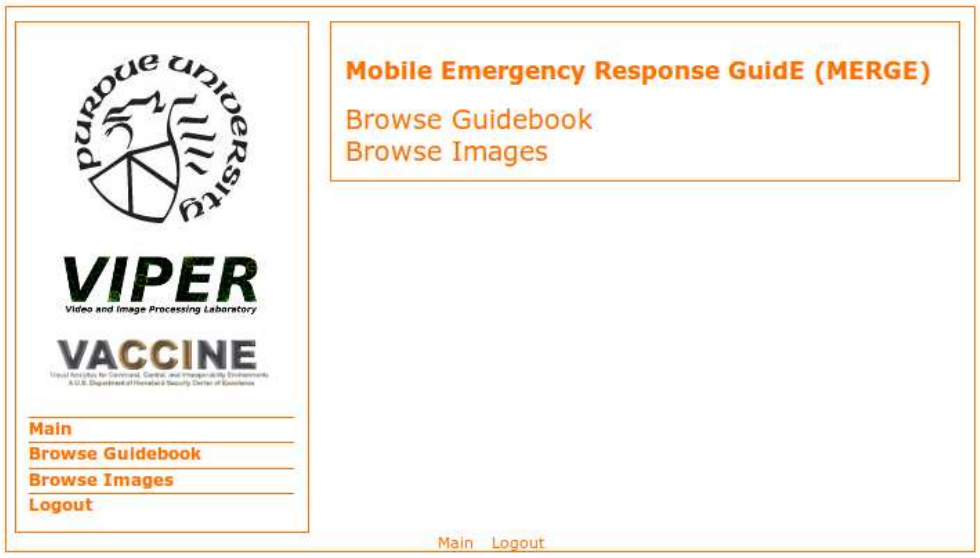


Fig. 4.35.: "Internal" Section of Desktop MERGE.

Fig. 4.36.: Search Guidebook Pages by Color, Symbol, Class, or UNID

Fig. 4.37.: Browse Guidebook Page Results

The Guide Page in MERGE is very similar to the Guide Page in the Emergency Response Guidebook. It contains the Guide Page Number, Guide Page Name, Categories (Potential Hazards, Public Safety, Emergency Response, and Supplemental Information), Sub-categories (Fire or Explosion, Health, Protective Clothing, Evacuation, Spill or Leak, First Aid, etc) and each sub-category contains a bulleted list of details (Figure 4.38).

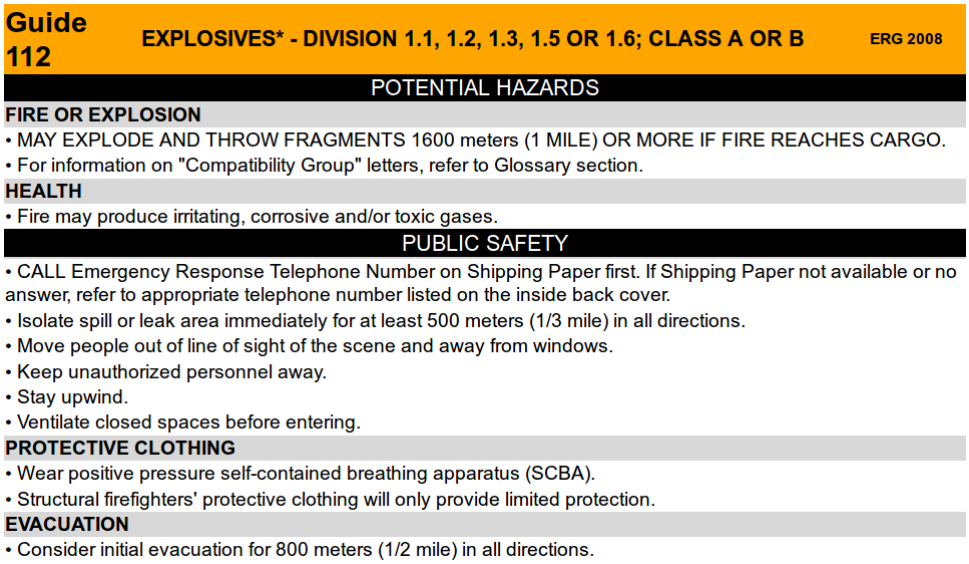


Fig. 4.38.: View Guidebook Page

Browse Images

Administrative users can browse images that have been uploaded (Figure 4.39). The images are listed along with the user who uploaded the image and the date and time they were taken. The user can choose between browsing the images containing signs (Signs) or the scene images uploaded for forensic analysis (Scene).

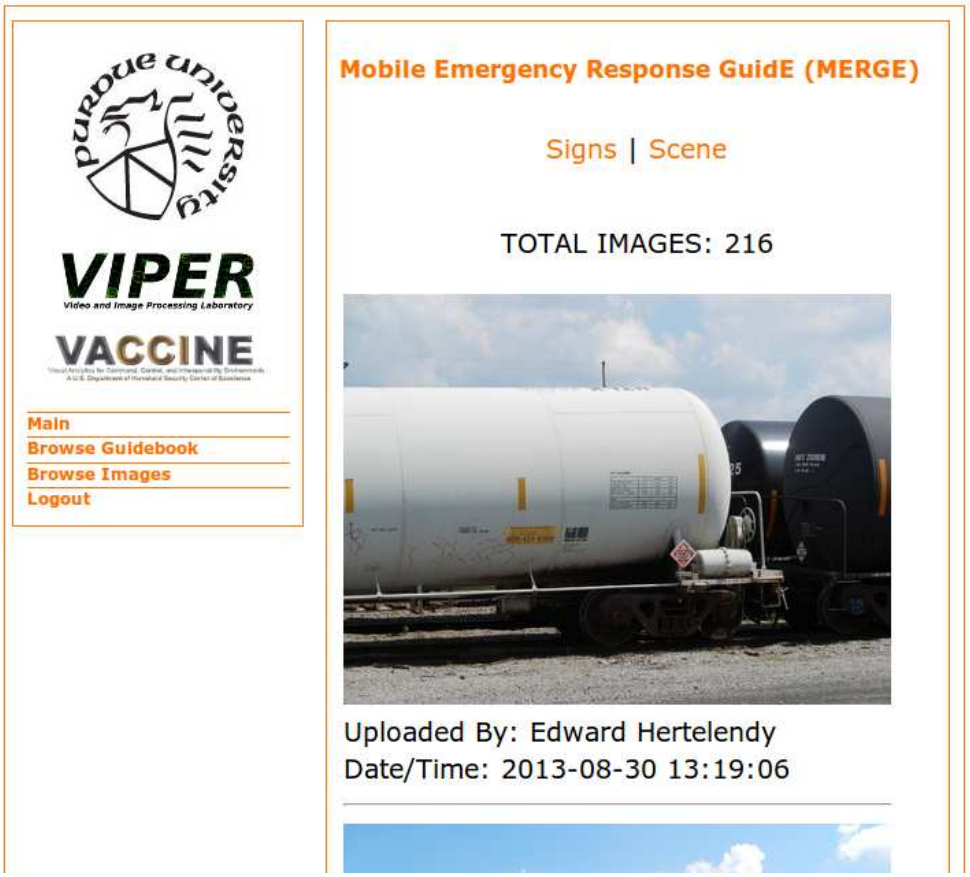


Fig. 4.39.: Browse Images

5. EXPERIMENTAL RESULTS

All the experiments in this section were done using a Samsung Galaxy Nexus mobile device with a dual-core 1.2GHz CPU and 1GB RAM for the client tasks, and a desktop computer with a quad-core 3.2GHz CPU and 32GB RAM for the server tasks.

5.1 GARI

5.1.1 RGB to Y'CH Conversion

In Section 3.4 and Appendix A we describe two approaches to transform the RGB color space to our Y'CH color space. The first, which we called *arithmetic approach*, converts RGB to Y'CH by only doing arithmetic operations The second, which we called *trigonometric approach*, converts RGB to YIQ color space as an intermediate step, and then to Y'CH, using arithmetic and trigonometric operations. As a reminder, Equation 5.1 shows the mathematical definition of the arithmetic approach and Equation 5.2 shows the mathematical definition of the trigonometric approach. Note that Equation 5.2 does not define the transformation RGB to YIQ, since it is a linear transformation, it will not have an influence on the execution time of the overall transformation RGB to Y'CH.

$$
\begin{aligned}
Y &= 0.299R + 0.587G + 0.114B \\
C &= max(R,G,B) - min(R,G,B) \\
&= M - n \\
H &= \begin{cases}
60(\frac{G-B}{C}) & \text{if M=R} \\
60(\frac{B-R}{C} + 2) & \text{if M=G} \\
60(\frac{R-G}{C} + 4) & \text{if M=B} \\
\text{undefined} & \text{if C=0}
\end{cases}
\end{aligned} \tag{5.1}
$$

$$
\begin{aligned}
Y &= 0.299R + 0.587G + 0.114B \\
C &= \sqrt{I^2 + Q^2} \\
H &= \begin{cases}
\arctan(\frac{Q}{I}) & I > 0 \\
\pi + \arctan(\frac{Q}{I}) & Q \geq 0,\ I < 0 \\
-\pi + \arctan(\frac{Q}{I}) & Q < 0,\ I < 0 \\
\frac{\pi}{2} & Q > 0,\ I = 0 \\
-\frac{\pi}{2} & Q < 0,\ I = 0 \\
\text{undefined} & Q = 0,\ I = 0
\end{cases}
\end{aligned} \tag{5.2}
$$

Given that trigonometric operations are computationally more complex than arithmetic operations [304], we could assume that the arithmetic approach is always computationally faster than the trigonometric approach. However, we conducted tests to verify this. Table 5.1 and Figure 5.1 show the results of both transformations using various number of data points on the HTC Desire. Note that each data point corresponds to a pixel operation. Also note that the functions used to compute the time differential both on the hand-held device are accurate to the nearest millisecond. One can see how the execution time of the trigonometric approach grows exponentially faster than the arithmetic approach when the number of data points is greater than

approximately one million. For example, for a five megapixel image (i.e., five million data points) the difference between the arithmetic approach and the trigonometric approach can be linearly interpolated to 3.36 seconds. Since the RGB to Y'CH conversion is done not only along a traced path during the color recognition process, but also on entire images during the image segmentation process, it is worth considering the arithmetic approach as a lightweight and fast approach if we plan on doing color image segmentation on the device in the future.

Table 5.1: Execution Time (seconds) of the Arithmetic and the Trigonometric Approaches For Color Conversion.

	Execution Time	
Data Points	**Arithmetic**	**Trigonometric**
100	0	0
1,000	0.002	0.004
10,000	0.010	0.010
100,000	0.02	0.10
1 million	0.20	0.96
10 million	1.91	9.39
100 million	18.37	91.85
1 billion	183	922

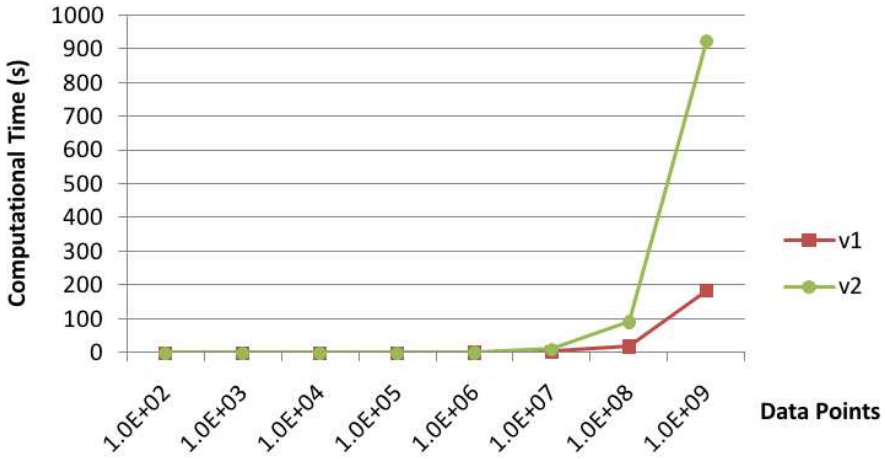


Fig. 5.1.: Execution Time with Respect to the Number of Data Points for the Arithmetic and the Trigonometric Approaches For Color Conversion.

5.1.2 Color Correction Based on Mobile Light Sensor[1]

To evaluate the performance of our proposed Color Correction Based on Mobile Light Sensor we did an experiment in 3 different scenarios: 1) using a fiducial marker in every image (M1), 2) using a fiducial marker every week (M2), 3) using the mobile light sensor values (M3). Using a fiducial marker every week means taking an image of the fiducial marker under daylight conditions to create a color correction matrix, and using this matrix on every image taken in the following week. For scenario M3 4,916 images were acquired during a period of three weeks during August of 2013, using a 5Mpx camera on a Samsung Galaxy Nexus mobile device, to obtain 612 unique lux values. Figure 5.2 illustrates the distribution of lux values for each lightning step.

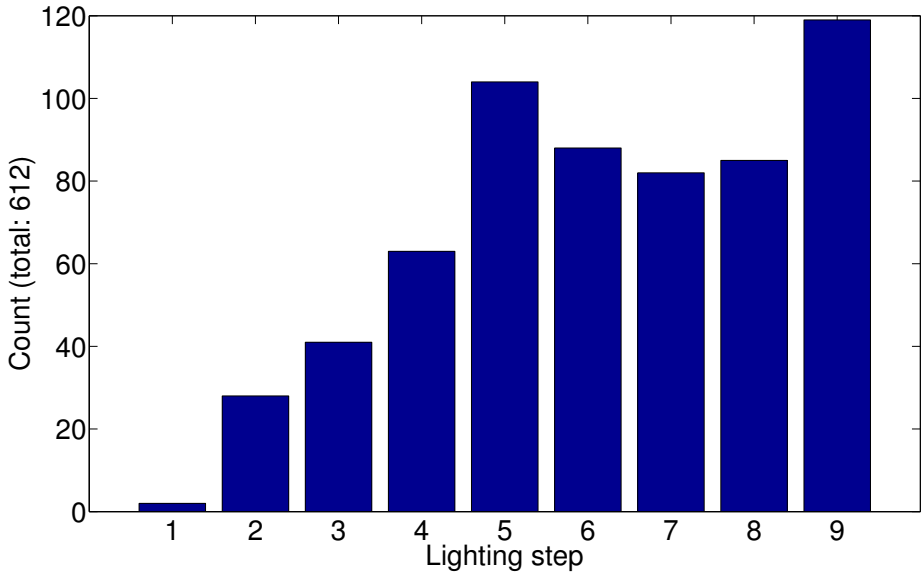


Fig. 5.2.: Distribution of Lux Values for Each Lightning Step.

For each scenario we computed 3 different color correction matrices to map colors under an unknown lighting condition and a D65 reference lighting condition: 1) CIELab based mapping ($M^{Lab}_{GT\rightarrow D65}$), 2) Linear-RGB mapping ($M^{RGB}_{GT\rightarrow D65}$), and 3) Polynomial-RGB mapping ($M^{RGBPOL}_{GT\rightarrow D65}$). The mapping $M^{Lab}_{GT\rightarrow D65}$ is described in

[1]The work presented in this section is partly based on the work by Chang Xu on color correction.

Section 3.3. The mappings $M^{RGB}_{GT\to D65}$ and $M^{RGBPOL}_{GT\to D65}$ are obtained following the description from [61]:

$$M^{Lab}_{GT\to D65} = \underset{M_{3\times 3}}{\operatorname{argmin}} \sum_{i=1}^{11} \left\| (Lab_i)^T_{D65} - M_{3\times 3}(Lab_i)^T_{GT} \right\| \tag{5.3}$$

$$M^{RGB}_{GT\to D65} = \underset{M_{3\times 3}}{\operatorname{argmin}} \sum_{i=1}^{11} \left\| (RGB_i)^T_{D65} - M_{3\times 3}(RGB_i)^T_{GT} \right\| \tag{5.4}$$

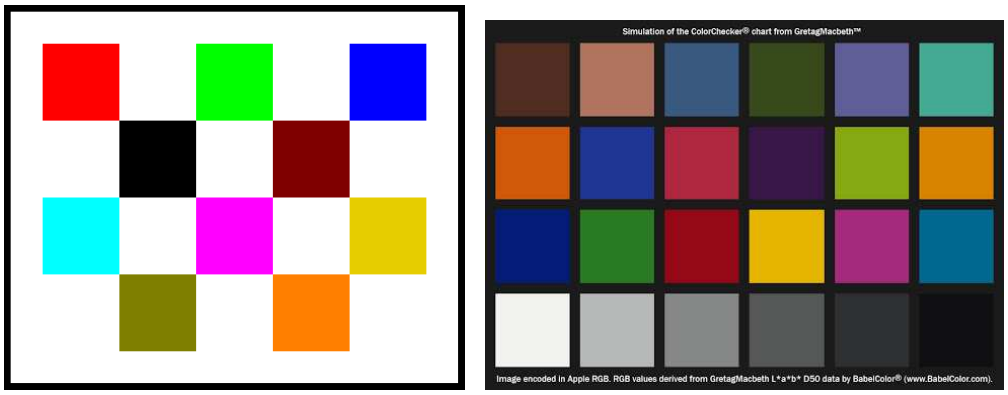
$$M^{RGBPOL}_{GT\to D65} = \underset{M_{3\times 10}}{\operatorname{argmin}} \sum_{i=1}^{11} \left\| (RGB_i)^T_{D65} - M_{3\times 10}P_{10\times 11} \right\|, \tag{5.5}$$

where

$$P_{10\times 11} = [R_{GT}\ G_{GT}\ B_{GT}\ R^2_{GT}\ G^2_{GT}\ B^2_{GT}\ R_{GT}B_{GT}\ R_{GT}G_{GT}\ G_{GT}B_{GT}\ 1]^T. \tag{5.6}$$

For this experiment we acquired 200 images during a period of 3 days during March of 2014 using a 8Mpx camera on a LG Nexus 5 mobile device. Each image contained the fiducial marker already introduced in Section 3.3 and a GregtagMacbeth Colorchecker [305], which is a calibrated color reference chart. Figure 5.3 shows both markers. The fiducial marker was used to obtain the color correction matrices in M1 and M2, and the GregtagMacbeth Colorchecker was used to compute the differences between the original image and the corrected images.

Each image was color corrected using the 3 mappings under each of the 3 scenarios for a total of 9 different color corrections. Figure 5.4 shows an example of color correction for each mapping. For each color corrected image we obtained the mean RGB channel errors Δ by calculating the Euclidean distances of the average color of each color patch in the GregtagMacbeth Colorchecker between the color corrected marker (RGB_{corr}) and the known reference marker under D65 illumination (RGB_{D65}). That is,

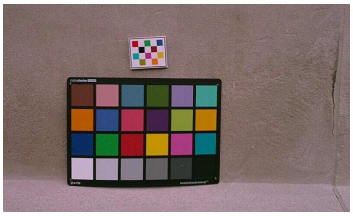


(a) Fiducial Marker (b) GregtagMacbeth Colorchecker

Fig. 5.3.: Fiducial Marker (left) and GregtagMacbeth Colorchecker (right).

$$\Delta = \frac{1}{24}\sum_{i=1}^{24}\left\|(RGB_i)^T_{corr} - (RGB_i)^T_{D65}\right\|. \tag{5.7}$$

(a) Original Image. Lux: 2219



(b) M1 Lab

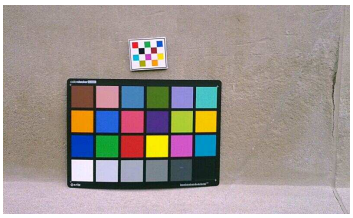
(c) M1 RGB

(d) M1 RGB POL

(e) M2 Lab

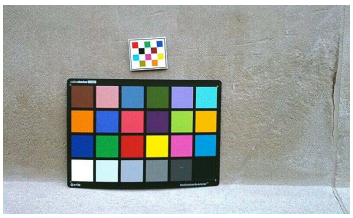
(f) M2 RGB

(g) M2 RGB POL



(h) M3 Lab

(i) M3 RGB



(j) M3 RGB POL

Fig. 5.4.: Color Correction Example Under Each Scenario and Each Mapping. M1: using a fiducial marker in every image, M2: using a fiducial marker every week, M3: using the mobile light sensor value.

Table 5.2 shows the mean RGB channel errors (Δ) and running times for each scenario (M1, M2, M3) and each mapping (Lab, RGB, RGB POL), including individual

errors in the R, G, and B color channels. We also include the Δ of image before correction for comparison. Figures 5.5 and 5.6 illustrate the RGB results in bar graphs. Note that since the errors are computed in the RGB color space, the Lab corrected images are transformed back to RGB. The time spent on this transformation is not taken into account in the running time.

Table 5.2: Mean Channel Errors (Δ) and Average Running Times (seconds) For Each Scenario (M1, M2, M3) and Mapping (Lab, RGB, RGB POL).

	RGB	**R**	**G**	**B**	**Time**
Before	14.06	6.90	6.10	8.35	-
M1 Lab	8.55	2.92	5.06	5.11	1.81
M1 RGB	11.99	4.49	4.97	8.07	1.11
M1 RGB POL	8.73	3.44	4.07	5.26	1.33
M2 Lab	12.72	6.02	5.97	7.18	1.80
M2 RGB	13.96	6.03	5.65	9.04	1.07
M2 RGB POL	12.18	5.84	5.04	6.82	1.31
M3 Lab	10.88	4.99	5.62	6.00	1.76
M3 RGB	13.27	5.63	5.33	8.59	1.05
M3 RGB POL	10.88	5.17	4.75	6.30	1.27

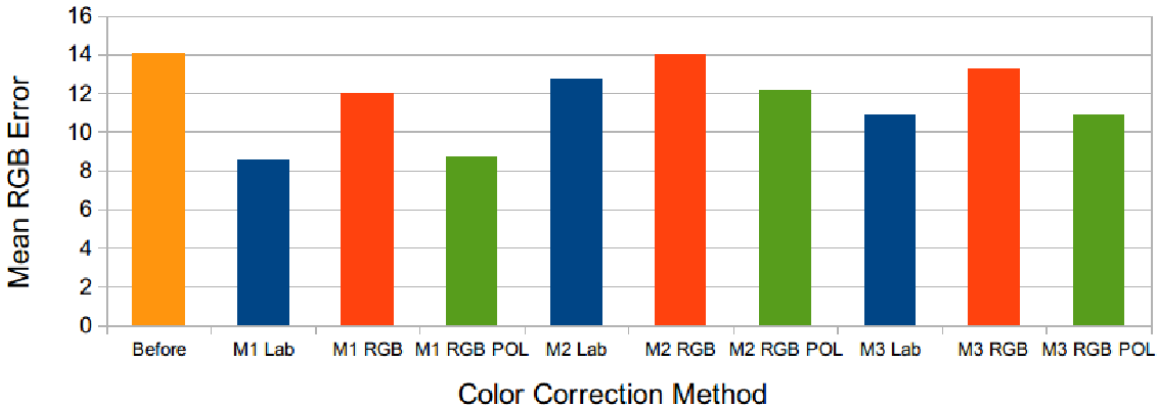


Fig. 5.5.: Mean Channel Errors (Δ) For Each Scenario (M1, M2, M3) and Mapping (Lab, RGB, RGB POL).

The Lab color correction method always gives the best results, at the expense of a small increase on the computational time. As expected, color correcting an image

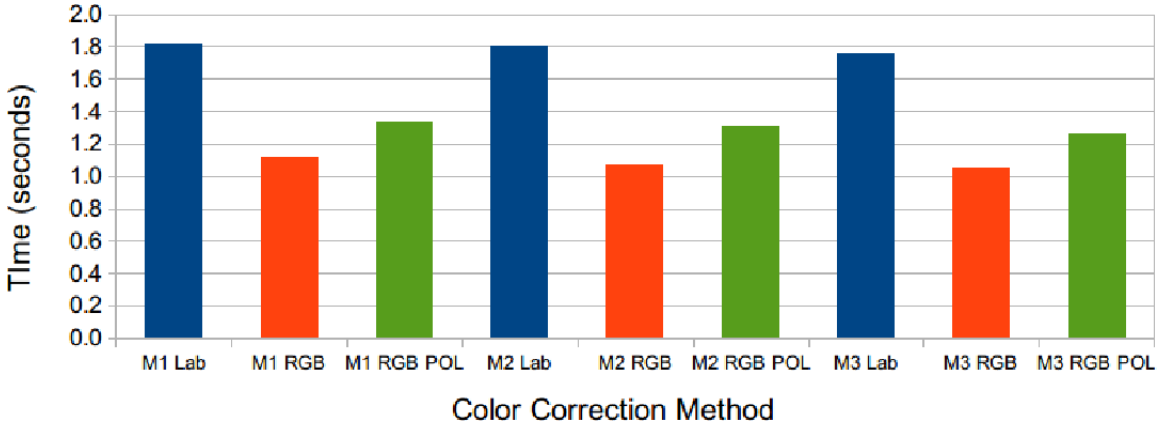


Fig. 5.6.: Average Running Times For Each Scenario (M1, M2, M3) and Mapping (Lab, RGB, RGB POL).

using always a fiducial marker produces the best results ($\Delta = 8.55$). However, the color correction based on the mobile light sensor produces better results than using a fiducial marker once a week ($\Delta = 10.88$ and $\Delta = 12.72$ respectively). Also, when using light sensor values we do not have to compute the color correction matrix for each image, thus being the fastest of the three scenarios.

5.1.3 Content Based Image Retrieval[2]

We did two experiments to determine the accuracy and the speed of our image retrieval approach.

The goal of the first experiment was to match query images to images in our database based on the scene. We call this process "Gang Graffiti Scene Recognition". That is, by finding features not only from the graffiti in the image, but also of the background. We trained 1,329 images from our database to extract a total of 633,764 SIFT descriptors (an average of 477 descriptor per image), and used hierarchical k-means to create a vocabulary tree. Figure 5.7 shows some samples from the training dataset. A separate set of 156 images was used for testing. Both training and testing images were acquired using multiple cameras with different resolutions, at different distances, and lighting conditions over a period of 3 years.

Each of the test images corresponded to one of the scenes in our database, but under different viewpoint, rotation, and illumination, and using different camera makes and models. Figure 5.8 shows some samples from the testing dataset. For each test image we retrieved its 5 closest matches from the training set and we gave it a score from 5 to 0, 5 meaning that the matching image was ranked in first position and 0 meaning that there was no match in the top 5 results. We called this scoring method "weighted top-5 accuracy".

Table 5.3 summarizes the results of the first experiment using different combinations of k and n_w in the range $k \in [2 \ldots 1,000]$ and $n_w \in [100 \ldots 1,000,000]$. Table

[2]The work presented in this section was done by the author in cooperation with Bin Zhao and Joonsoo Kim.

Fig. 5.7.: Samples from Training Dataset.

Fig. 5.8.: Samples Image Matches. Left: Training Images (Samsung Galaxy Nexus). Right: Matching Testing Images (Casio PowerShot S95).

5.4 shows the Top-1 accuracies for the same ranges of k and n_w. Tables 5.5 and 5.6 show the average training and query times. Figures 5.9 to 5.11 illustrate the same information using color maps. Even though the retrieval accuracy increases with the number of leaves, the query time is directly related to the number of nodes and levels in the vocabulary tree, as shown in Figures 5.12 and 5.13. A wise choice for k and n_w would then take into account both the accuracy and the query time (not the training time, since it does not affect the real time retrieval). For $k = 3$ and $n_w = 10,000$ we obtain a retrieval accuracy of 99.10% with a Top-1 accuracy of 96.15% and an average query time of 70 ms. As a comparison, using basic L2-norm matching of SIFT features between two images in the same computer takes 0.18 seconds on average. Therefore, a query against the 1,329 training images takes 4 minutes on average.

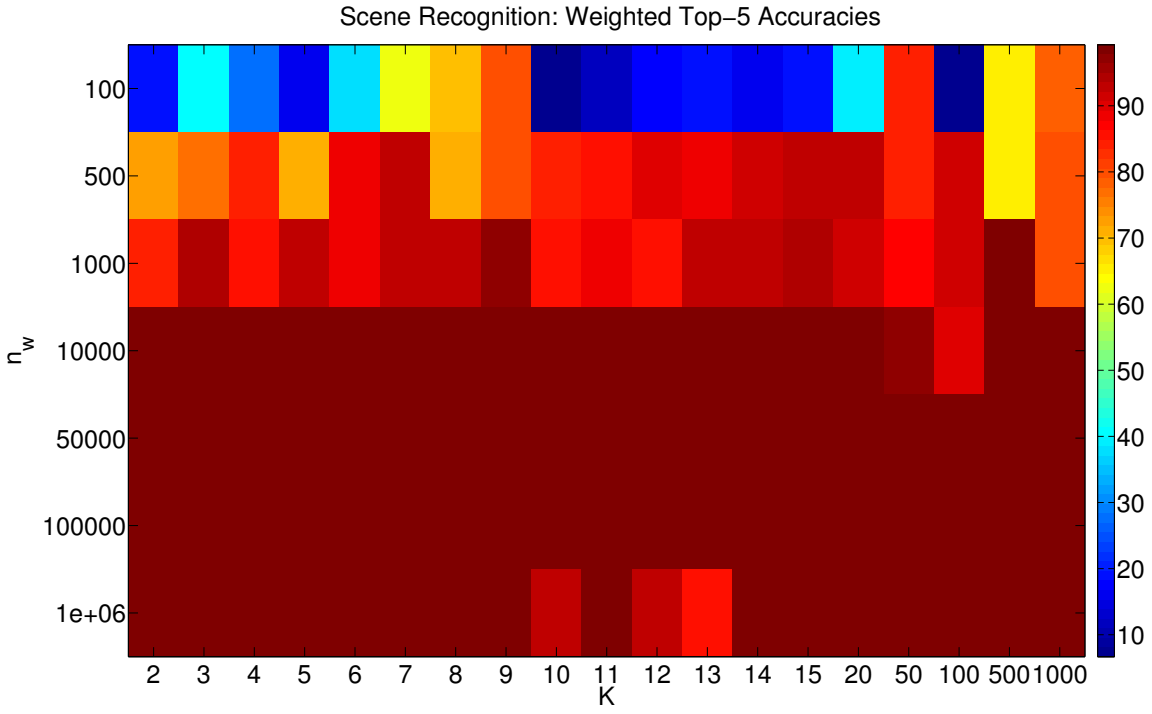


Fig. 5.9.: Color Map of Weighted Top-5 Accuracies of Scene Recognition Using Different Values of k and n_w.

Table 5.3: Weighted Top-5 Accuracies of Scene Recognition for Different Values of k and n_w (percentage).

n_w \ k	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
100	19.62	40.64	28.21	16.15	38.33	62.82	68.97	79.23	7.56	11.54
500	72.31	76.41	84.74	70.77	88.72	92.05	71.15	79.62	83.85	85.51
1000	84.74	94.23	84.87	93.21	88.59	92.18	92.95	96.54	85.00	87.69
10000	99.10	99.10	98.21	98.21	98.46	98.21	98.72	98.97	98.21	97.95
50000	98.85	99.10	98.85	98.85	99.23	99.10	99.10	98.85	99.10	99.10
100000	98.85	98.85	98.97	99.10	99.23	98.97	99.23	99.23	99.10	99.10
1000000	99.10	98.85	98.85	98.33	99.23	98.97	97.95	99.23	92.18	99.23

n_w \ k	**12**	**13**	**14**	**15**	**20**	**50**	**100**	**500**	**1000**
100	17.69	19.10	15.64	18.46	39.23	83.33	6.67	65.64	78.46
500	89.74	87.69	90.64	92.69	92.95	83.59	91.80	65.64	80.26
1000	85.51	92.31	92.31	94.49	91.28	87.31	91.28	98.46	80.00
10000	98.33	98.72	98.46	98.59	98.72	96.54	90.26	98.33	98.72
50000	99.23	99.10	99.23	98.97	98.97	98.21	98.72	98.72	98.33
100000	99.10	99.10	99.10	99.23	98.46	98.46	98.33	97.95	99.10
1000000	92.31	85.00	99.10	98.97	98.72	99.23	98.21	99.10	99.10

Table 5.4: Top-1 Accuracies of Scene Recognition for Different Values of k and n_w (percentage).

n_w \ k	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
100	10.90	17.31	10.90	6.41	14.74	37.82	49.36	58.97	3.21	4.49
500	52.56	55.13	69.23	41.03	76.28	82.05	51.28	58.97	71.80	75.64
1000	73.08	83.33	72.44	83.97	73.72	82.69	83.33	89.10	77.56	73.08
10000	73.08	96.15	93.59	94.23	94.23	93.59	94.23	95.51	91.67	93.59
50000	96.15	96.15	95.51	95.51	96.80	96.15	96.15	94.87	96.15	96.15
100000	93.59	93.59	95.51	96.15	96.80	95.51	96.80	96.80	96.15	96.15
1000000	96.15	96.15	95.51	95.51	94.23	96.80	95.51	95.51	96.80	96.80

n_w \ k	**12**	**13**	**14**	**15**	**20**	**50**	**100**	**500**	**1000**
100	9.62	10.26	8.33	9.62	19.23	63.46	1.28	42.31	62.18
500	80.13	76.92	82.05	82.69	80.77	66.67	76.92	39.74	62.18
1000	71.80	81.41	80.77	83.97	71.80	69.23	77.56	95.51	61.54
10000	92.31	94.23	95.51	94.23	94.23	92.31	78.21	95.51	96.15
50000	96.80	96.15	96.80	95.51	95.51	94.23	95.51	96.15	96.15
100000	96.15	96.15	96.15	96.80	94.87	94.87	95.51	95.51	96.15
1000000	96.80	94.23	94.23	92.31	94.87	96.80	96.15	96.15	96.15

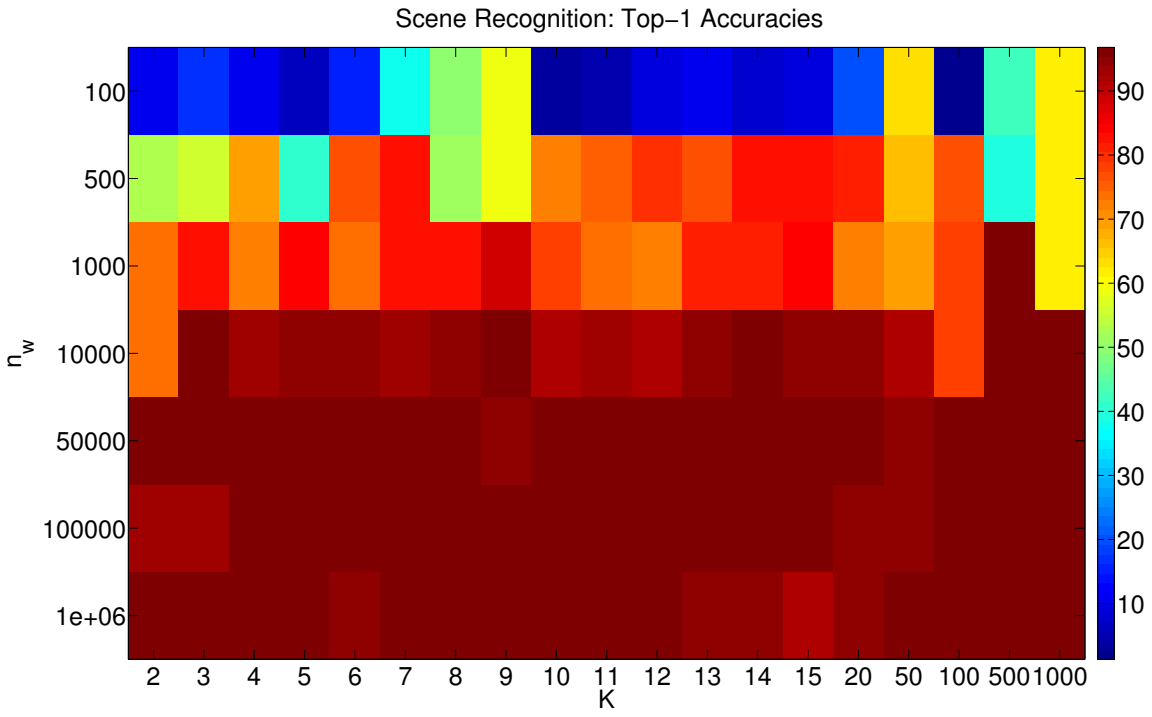


Fig. 5.10.: Color Map of Top-1 Accuracies of Scene Recognition Using Different Values of k and n_w.

Table 5.5: Training Times of Scene Recognition for Different Values of k and n_w (minutes).

n_w \ k	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
100	69	94	110	116	107	142	219	399	169	241
500	92	106	112	141	173	189	288	220	243	272
1000	87	121	122	179	186	169	234	256	250	343
10000	146	133	152	179	221	220	220	257	261	416
50000	152	134	152	170	202	208	218	256	302	293
100000	175	154	143	189	205	219	287	270	338	391
1000000	723	429	229	292	328	367	350	323	325	396

n_w \ k	**12**	**13**	**14**	**15**	**20**	**50**	**100**	**500**	**1000**
100	216	365	278	353	309	327	340	293	386
500	359	337	478	519	307	321	342	292	385
1000	448	373	367	452	308	328	346	300	385
10000	380	369	460	497	309	323	346	299	390
50000	327	371	401	374	309	328	346	300	392
100000	452	370	412	415	457	308	379	320	234
1000000	424	492	530	552	785	311	327	306	205

Table 5.6: Query Times of Scene Recognition for Different Values of k and n_w (seconds).

n_w \ k	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
100	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
500	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
1000	0.07	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
10000	3.45	0.07	0.07	0.07	0.07	0.07	0.07	0.07	0.07	0.07
50000	52.27	0.07	0.08	0.08	0.12	0.09	0.11	0.07	0.08	0.09
100000	199.09	54.46	35.68	0.15	0.12	0.09	0.11	0.14	0.08	0.09
1000000	6381.98	3444.07	3408.00	3378.00	3349.00	3325.00	3291.00	3278.00	3215.00	3211.00

n_w \ k	**12**	**13**	**14**	**15**	**20**	**50**	**100**	**500**	**1000**
100	0.06	0.06	0.06	0.06	0.06	0.07	0.07	0.09	0.12
500	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.09	0.12
1000	0.06	0.06	0.06	0.06	0.07	0.07	0.07	0.14	0.12
10000	0.07	0.07	0.07	0.07	0.08	0.08	0.07	0.14	0.21
50000	0.10	0.11	0.15	0.07	0.08	0.08	0.18	0.14	0.20
100000	0.10	0.11	0.12	0.14	0.08	0.08	0.13	0.14	0.20
1000000	3207.00	3185.00	3182.00	3171.00	3122.00	3081.00	3051.00	3036.00	2997.00

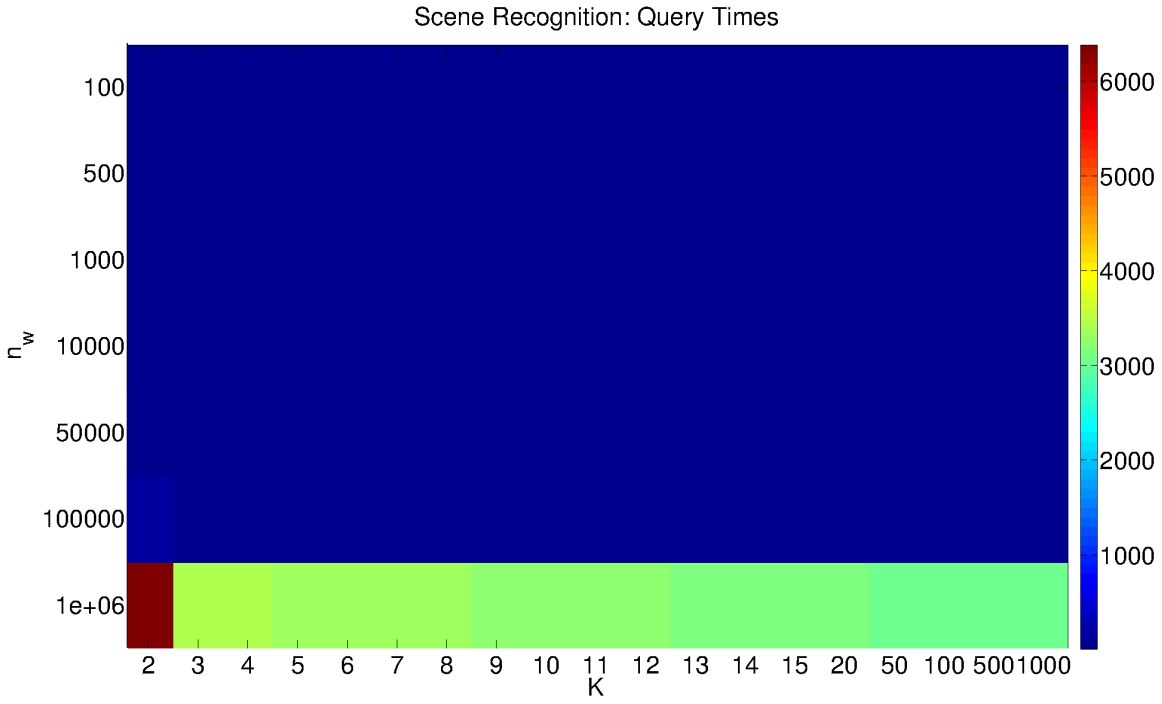


Fig. 5.11.: Color Map of Query Times of Scene Recognition Using Different Values of k and n_w.

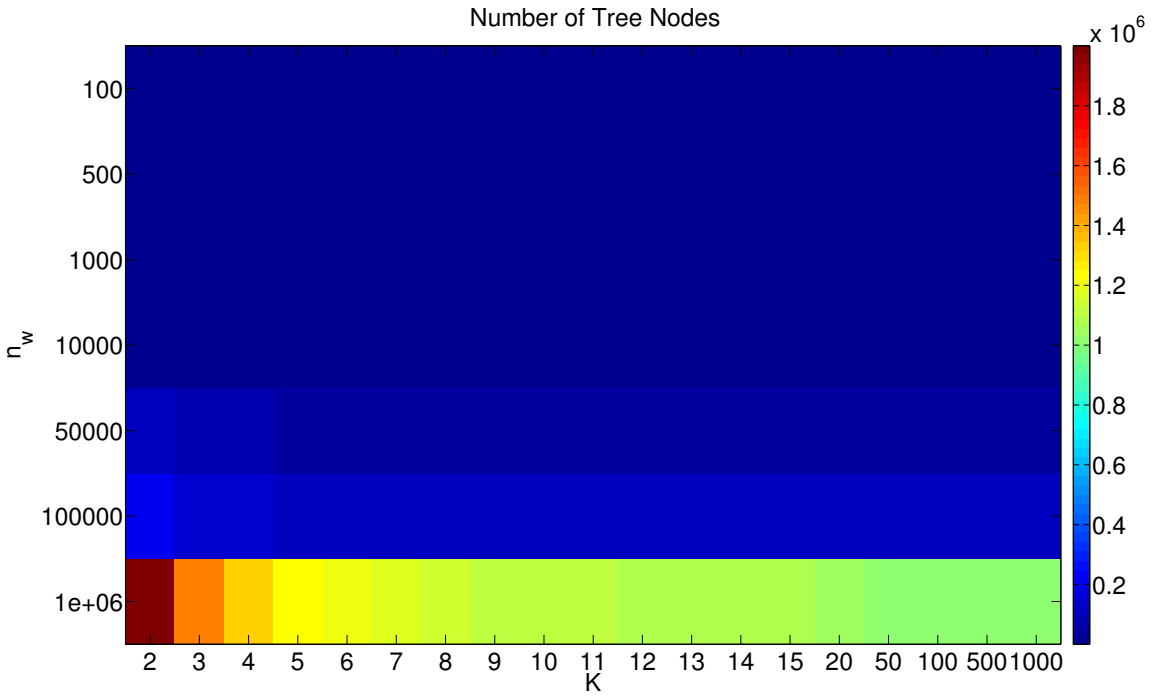


Fig. 5.12.: Number of Vocabulary Tree Nodes As a Function of k and n_w.

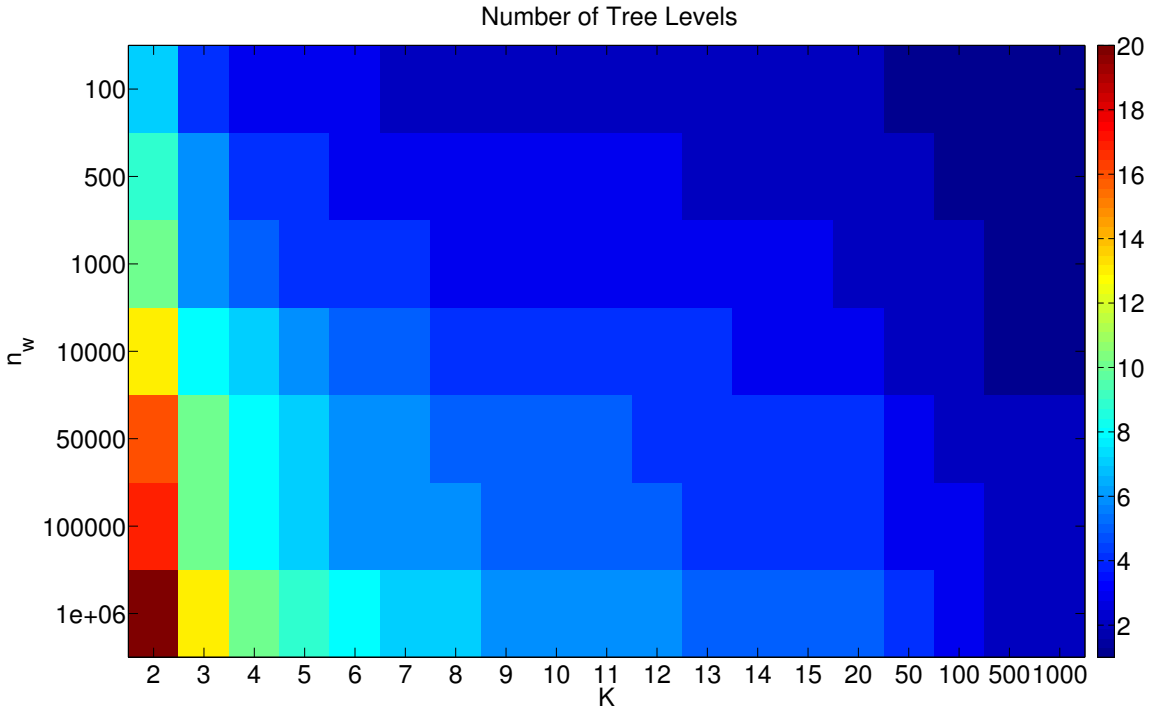


Fig. 5.13.: Number of Vocabulary Tree Levels As a Function of k and n_w.

It is worth noting that although this experiment only accounted for scene recognition, we found that sometimes the results returned included scenes of nearby graffiti or even graffiti that have been removed. Figure 5.14 illustrates an example.

Fig. 5.14.: Query Images (Left) And Similar Retrieved Scenes (Right).

The goal of the second experiment was to classify query images into categories based on a set of gang graffiti symbols. We call this process "Gang Graffiti Component Classification". We created 14 classes for training, where each class corresponds to a distinct graffiti component, including: *0*, *1*, *2*, *3*, *4*, *5-point star*, *G*, *6-point star*, *8*, *arrow*, *E*, *pitchfork*, *S*, and *X*. For each class we trained 17 images, making a total of 238 images for training. Each training image consists of one graffiti component in black with white background. A separate set of 56 images, 4 images per class, was used for testing. Each of the test images also consisted of one graffiti component in black with white background. Figure 5.15 shows some sample images. Note the inter-class variance as well as the intra-class similarity.

Since in this experiment we used our proposed SIFT-based Local Shape Context (LSC) descriptors to generate the vocabulary tree we need to set two additional parameters: n_r for the number of concentric circumferences representing log-radial distance bins and n_θ for the number of angular bins. Given the results of the first experiment we chose $k = 3$ and $n_w = 10,000$ to create the vocabulary tree.

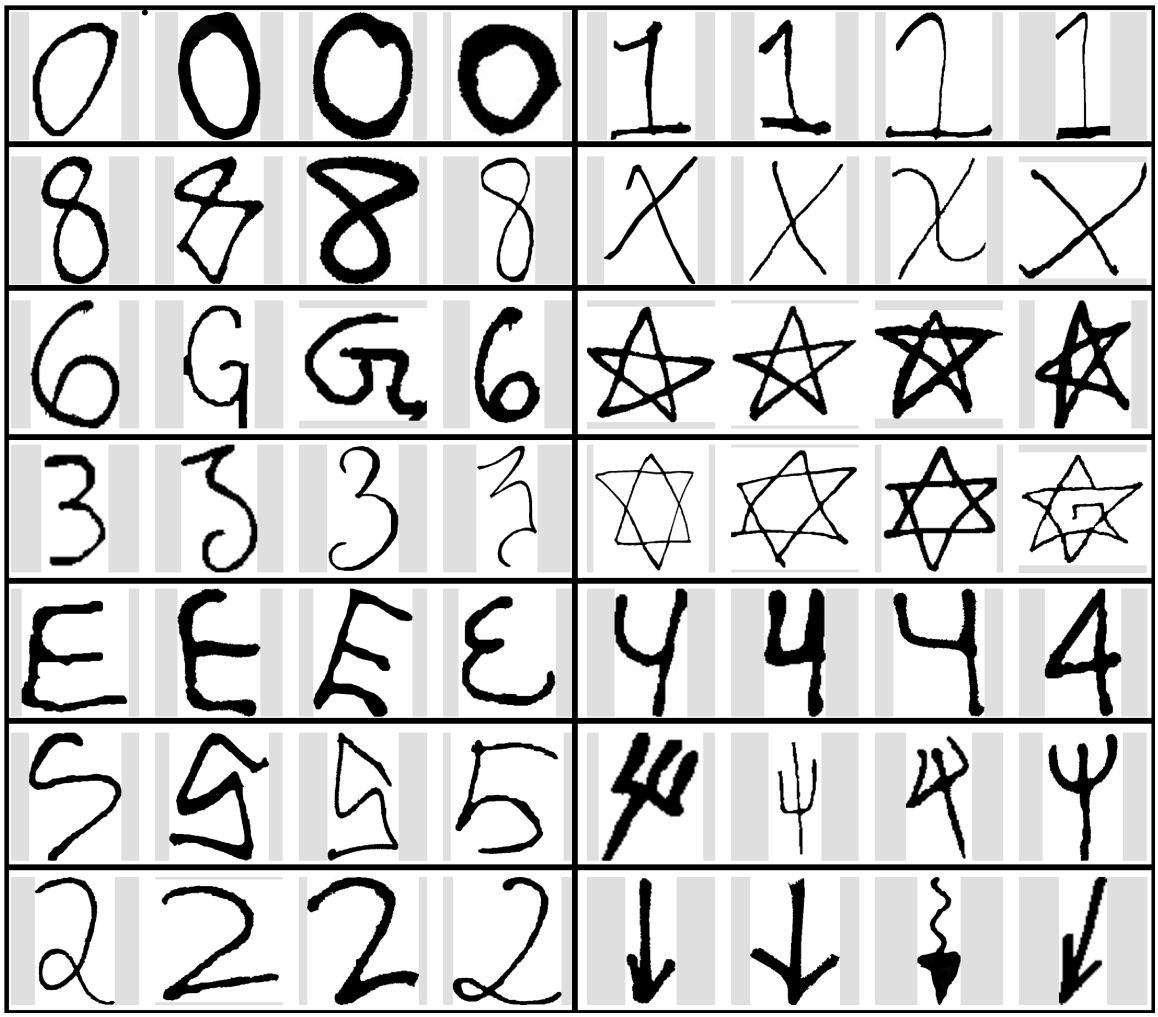


Fig. 5.15.: Sample Images for Each Class. From left to right, top to bottom, in groups of 4 images: *0*, *1*, *8*, *X*, *G*, *5-point star*, *3*, *6-point star*, *E*, *4*, *S*, *pitchfork*, *2*, and *arrow*. Note the inter-class variance as well as the intra-class similarity.

For each query image we retrieved its 10 closest matches from the training set and we assigned a class based on the following scoring method. Given the scores (votes) of the 10 closest matches $p = \{p_1, \ldots, p_{10}\}$ in ascending order, we manually group them into N classes, $N \in \{1, \ldots, 14\}$. We add up the new scores associated to each class, and we assign the class C with the highest score to the query image, such that $C = \underset{n}{\text{argmax}}\{\sum_k p_k^{(n)}\}$, where k is the set of indices of p belonging to the n-th class, $n \in \{1, \ldots, N\}$.

Tables 5.7 and 5.8 summarize the results of the second experiment using different combinations of n_r and n_θ in the range $n_r \in [1 \ldots 20]$ and $n_\theta \in [4 \ldots 30]$. Tables 5.9 and 5.10 show the Top-10 accuracies, and Tables 5.11 and 5.12 show the Top-5 accuracies for the same ranges of n_r and n_θ.

Figures 5.16 to 5.18 illustrate the same information using color maps. Low values of n_θ cause low classification accuracy, because we do not have enough discrimination between feature locations. High values of both n_r and n_θ also cause low classification accuracy, because we do not account for the elasticity of the graffiti components.

Since we use fixed values of k and n_w on this experiment, n_r and n_θ do not have a strong impact in the query time. Therefore we can choose our values from the results of Tables 5.7 and 5.8. For $n_r = 3$ and $n_\theta = 16$ we achieve a classification accuracy of 89.29% with a Top-10 accuracy of 94.64% and a Top-5 accuracy of 92.86%. The average query time is 71 ms, from which 6 ms are spent on average to compute the LSC descriptor.

Figure 5.19 illustrates the confusion matrix [306] for each of the 14 classes when $n_r = 3$ and $n_\theta = 16$. Each column of the matrix represents the instances in a predicted class, and each row represents the instances in the ground-truth (i.e. expected) class. Hight counts on the diagonal indicate high classification accuracy for a specific class. Table 5.13 summarizes the classification results for each class, including precision, recall and F_1 score for each class [307]. Given a confusion matrix M where the x-axis

Table 5.7: Classification Accuracies of Gang Graffiti Component Classification for $n_r \in [1 \dots 20]$ and $n_\theta \in [4 \dots 17]$ (percentage).

n_r \ n_θ	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**
1	55.36	69.64	71.43	75.00	75.00	73.21	76.79	71.43	82.14	78.57	80.36	71.43	89.29	80.36
2	60.71	67.86	76.79	69.64	73.21	82.14	73.21	80.36	78.57	78.57	82.14	85.71	87.50	87.50
3	64.29	69.64	80.36	76.79	75.00	80.36	78.57	82.14	76.79	76.79	85.71	78.57	89.29	82.14
4	71.43	71.43	73.21	76.79	82.14	78.57	80.36	80.36	76.79	78.57	78.57	78.57	78.57	83.93
5	71.43	76.79	76.79	73.21	78.57	83.93	76.79	82.14	80.36	80.36	80.36	80.36	83.93	82.14
6	75.00	75.00	75.00	76.79	76.79	69.64	75.00	75.00	76.79	73.21	76.79	78.57	82.14	85.71
7	67.86	73.21	71.43	73.21	67.86	80.36	78.57	73.21	76.79	73.21	71.43	80.36	83.93	80.36
8	62.50	75.00	71.43	76.79	76.79	80.36	73.21	80.36	82.14	76.79	80.36	80.36	80.36	80.36
9	66.07	73.21	73.21	76.79	73.21	82.14	76.79	80.36	75.00	83.93	75.00	80.36	82.14	78.57
10	69.64	75.00	71.43	75.00	76.79	78.57	76.79	76.79	78.57	82.14	73.21	78.57	78.57	83.93
11	78.57	73.21	71.43	75.00	75.00	76.79	76.79	75.00	83.93	83.93	78.57	80.36	83.93	76.79
12	76.79	75.00	71.43	78.57	76.79	78.57	76.79	80.36	83.93	82.14	75.00	82.14	82.14	80.36
13	69.64	71.43	78.57	78.57	80.36	76.79	69.64	80.36	78.57	82.14	76.79	78.57	76.79	80.36
14	69.64	69.64	76.79	76.79	76.79	80.36	78.57	76.79	80.36	78.57	82.14	78.57	82.14	82.14
15	67.86	71.43	76.79	71.43	80.36	75.00	80.36	76.79	82.14	78.57	78.57	85.71	78.57	76.79
16	71.43	69.64	71.43	75.00	73.21	73.21	75.00	80.36	80.36	82.14	75.00	80.36	75.00	83.93
17	66.07	69.64	75.00	73.21	73.21	75.00	78.57	78.57	80.36	78.57	75.00	80.36	75.00	80.36
18	67.86	75.00	73.21	69.64	78.57	80.36	78.57	78.57	78.57	78.57	82.14	78.57	82.14	78.57
19	67.86	69.64	71.43	78.57	78.57	76.79	75.00	76.79	76.79	80.36	76.79	76.79	78.57	75.00
20	64.29	75.00	73.21	80.36	80.36	78.57	67.86	80.36	73.21	76.79	76.79	78.57	85.71	80.36

Table 5.8: Classification Accuracies of Gang Graffiti Component Classification for $n_r \in [1 \ldots 20]$ and $n_\theta \in [18 \ldots 30]$ (percentage).

n_r \ n_θ	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**
1	82.14	80.36	82.14	85.71	82.14	87.50	80.36	82.14	80.36	78.57	78.57	80.36	78.57
2	83.93	83.93	83.93	83.93	83.93	80.36	83.93	85.71	83.93	80.36	82.14	78.57	80.36
3	82.14	82.14	82.14	87.50	78.57	82.14	82.14	83.93	78.57	76.79	82.14	78.57	78.57
4	87.50	76.79	82.14	82.14	85.71	80.36	85.71	78.57	82.14	75.00	76.79	82.14	78.57
5	85.71	83.93	85.71	82.14	87.50	82.14	80.36	80.36	80.36	80.36	85.71	75.00	78.57
6	85.71	78.57	85.71	82.14	85.71	85.71	82.14	80.36	80.36	78.57	80.36	82.14	76.79
7	82.14	78.57	78.57	82.14	85.71	87.50	76.79	78.57	80.36	85.71	76.79	80.36	76.79
8	80.36	76.79	85.71	78.57	76.79	80.36	78.57	80.36	85.71	83.93	78.57	82.14	82.14
9	76.79	76.79	82.14	78.57	76.79	78.57	82.14	75.00	78.57	71.43	80.36	82.14	82.14
10	82.14	80.36	83.93	80.36	78.57	76.79	78.57	78.57	75.00	80.36	75.00	75.00	78.57
11	80.36	78.57	83.93	82.14	78.57	75.00	76.79	78.57	78.57	75.00	76.79	78.57	80.36
12	80.36	78.57	83.93	82.14	80.36	78.57	80.36	82.14	73.21	78.57	80.36	78.57	80.36
13	78.57	78.57	80.36	76.79	82.14	78.57	78.57	82.14	75.00	80.36	76.79	78.57	82.14
14	75.00	80.36	82.14	80.36	73.21	75.00	82.14	80.36	73.21	82.14	76.79	69.64	82.14
15	80.36	75.00	82.14	80.36	82.14	78.57	80.36	83.93	80.36	78.57	76.79	80.36	69.64
16	80.36	80.36	78.57	76.79	75.00	80.36	76.79	78.57	76.79	80.36	80.36	76.79	78.57
17	76.79	78.57	80.36	76.79	80.36	76.79	75.00	83.93	82.14	73.21	71.43	76.79	78.57
18	73.21	82.14	80.36	67.86	78.57	76.79	78.57	75.00	73.21	78.57	82.14	76.79	71.43
19	80.36	80.36	82.14	82.14	78.57	83.93	73.21	78.57	80.36	76.79	75.00	78.57	76.79
20	78.57	75.00	78.57	76.79	69.64	78.57	76.79	76.79	67.86	82.14	78.57	82.14	76.79

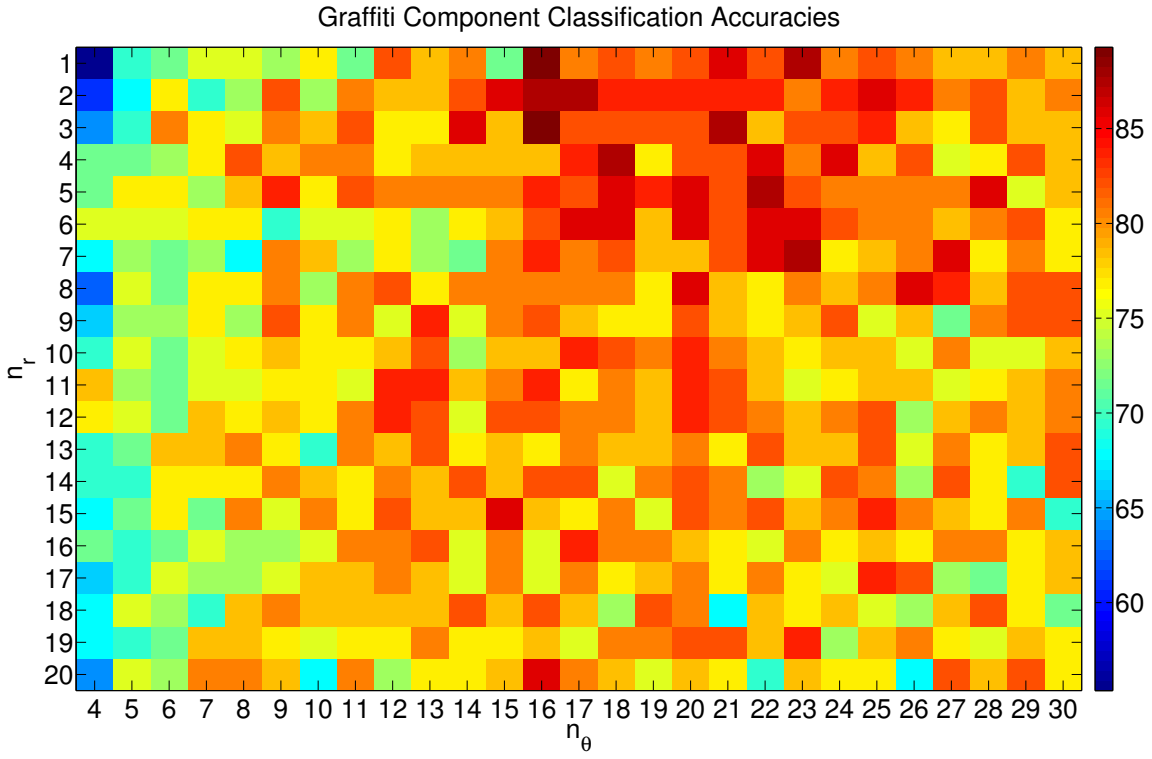


Fig. 5.16.: Color Map of Classification Accuracies of Gang Graffiti Component Classification Using Different Values of n_r and n_θ.

Table 5.9: Top-10 Classification Accuracies of Gang Graffiti Component Classification for $n_r \in [1 \ldots 20]$ and $n_\theta \in [4 \ldots 17]$ (percentage).

n_r \ n_θ	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**
1	91.07	92.86	96.43	98.21	96.43	98.21	100.00	100.00	96.43	98.21	96.43	98.21	98.21	96.43
2	87.50	92.86	100.00	98.21	98.21	94.64	98.21	94.64	94.64	94.64	94.64	96.43	96.43	98.21
3	96.43	98.21	96.43	98.21	94.64	96.43	96.43	96.43	94.64	96.43	92.86	98.21	94.64	98.21
4	92.86	96.43	98.21	94.64	94.64	98.21	94.64	94.64	98.21	98.21	96.43	94.64	94.64	100.00
5	92.86	96.43	96.43	96.43	94.64	94.64	92.86	94.64	94.64	92.86	92.86	96.43	96.43	96.43
6	91.07	91.07	96.43	98.21	98.21	96.43	96.43	96.43	98.21	94.64	94.64	100.00	100.00	96.43
7	92.86	92.86	96.43	92.86	96.43	94.64	89.29	91.07	96.43	96.43	98.21	94.64	94.64	96.43
8	87.50	92.86	94.64	92.86	94.64	96.43	94.64	94.64	96.43	96.43	94.64	92.86	96.43	92.86
9	92.86	92.86	98.21	92.86	94.64	94.64	94.64	92.86	92.86	96.43	96.43	98.21	94.64	96.43
10	96.43	92.86	91.07	91.07	96.43	98.21	92.86	92.86	94.64	92.86	96.43	91.07	98.21	94.64
11	96.43	89.29	92.86	92.86	94.64	96.43	96.43	94.64	96.43	96.43	94.64	96.43	96.43	92.86
12	92.86	91.07	94.64	92.86	92.86	94.64	94.64	98.21	96.43	96.43	96.43	94.64	96.43	98.21
13	91.07	92.86	94.64	96.43	96.43	92.86	91.07	96.43	96.43	94.64	96.43	96.43	94.64	94.64
14	89.29	91.07	94.64	98.21	96.43	92.86	92.86	92.86	91.07	96.43	94.64	94.64	94.64	94.64
15	92.86	87.50	96.43	94.64	92.86	94.64	91.07	92.86	92.86	96.43	96.43	96.43	92.86	92.86
16	92.86	91.07	96.43	94.64	94.64	91.07	91.07	96.43	94.64	96.43	94.64	94.64	96.43	94.64
17	91.07	87.50	92.86	96.43	94.64	96.43	89.29	96.43	91.07	98.21	92.86	98.21	98.21	94.64
18	96.43	87.50	96.43	91.07	94.64	91.07	92.86	94.64	96.43	92.86	96.43	96.43	94.64	96.43
19	91.07	91.07	92.86	98.21	98.21	91.07	92.86	92.86	94.64	91.07	96.43	94.64	94.64	96.43
20	92.86	91.07	91.07	94.64	92.86	94.64	94.64	96.43	96.43	94.64	96.43	92.86	96.43	100.00

Table 5.10: Top-10 Classification Accuracies of Gang Graffiti Component Classification for $n_r \in [1 \ldots 20]$ and $n_\theta \in [18 \ldots 30]$ (percentage).

n_r \ n_θ	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**
1	96.43	98.21	98.21	100.00	98.21	98.21	96.43	98.21	100.00	96.43	100.00	98.21	100.00
2	98.21	96.43	98.21	100.00	100.00	98.21	98.21	98.21	98.21	96.43	98.21	100.00	96.43
3	96.43	94.64	96.43	98.21	100.00	100.00	98.21	100.00	98.21	98.21	100.00	100.00	98.21
4	100.00	98.21	96.43	98.21	96.43	94.64	96.43	98.21	96.43	96.43	94.64	96.43	94.64
5	100.00	96.43	94.64	94.64	96.43	98.21	98.21	96.43	94.64	96.43	98.21	94.64	96.43
6	98.21	91.07	94.64	96.43	100.00	96.43	94.64	96.43	94.64	96.43	96.43	98.21	92.86
7	98.21	96.43	96.43	96.43	96.43	94.64	94.64	96.43	98.21	96.43	92.86	98.21	96.43
8	98.21	96.43	96.43	96.43	96.43	96.43	96.43	98.21	94.64	96.43	96.43	98.21	98.21
9	98.21	96.43	94.64	94.64	96.43	98.21	94.64	98.21	98.21	96.43	98.21	92.86	94.64
10	96.43	96.43	98.21	96.43	98.21	96.43	96.43	92.86	96.43	98.21	94.64	94.64	94.64
11	98.21	98.21	94.64	96.43	98.21	98.21	96.43	94.64	92.86	96.43	96.43	94.64	96.43
12	100.00	98.21	94.64	96.43	91.07	96.43	96.43	96.43	96.43	96.43	98.21	98.21	96.43
13	94.64	96.43	92.86	96.43	98.21	96.43	96.43	96.43	96.43	94.64	91.07	96.43	98.21
14	94.64	100.00	94.64	100.00	96.43	98.21	96.43	94.64	96.43	94.64	92.86	92.86	98.21
15	96.43	96.43	98.21	100.00	98.21	94.64	96.43	96.43	96.43	98.21	94.64	96.43	96.43
16	96.43	96.43	100.00	92.86	96.43	96.43	98.21	94.64	98.21	96.43	94.64	96.43	98.21
17	96.43	92.86	94.64	92.86	98.21	94.64	94.64	96.43	98.21	96.43	96.43	96.43	96.43
18	94.64	98.21	100.00	94.64	100.00	96.43	96.43	96.43	96.43	94.64	94.64	94.64	92.86
19	100.00	94.64	96.43	96.43	96.43	91.07	96.43	96.43	98.21	96.43	94.64	96.43	94.64
20	96.43	98.21	98.21	98.21	98.21	96.43	96.43	92.86	96.43	94.64	98.21	92.86	94.64

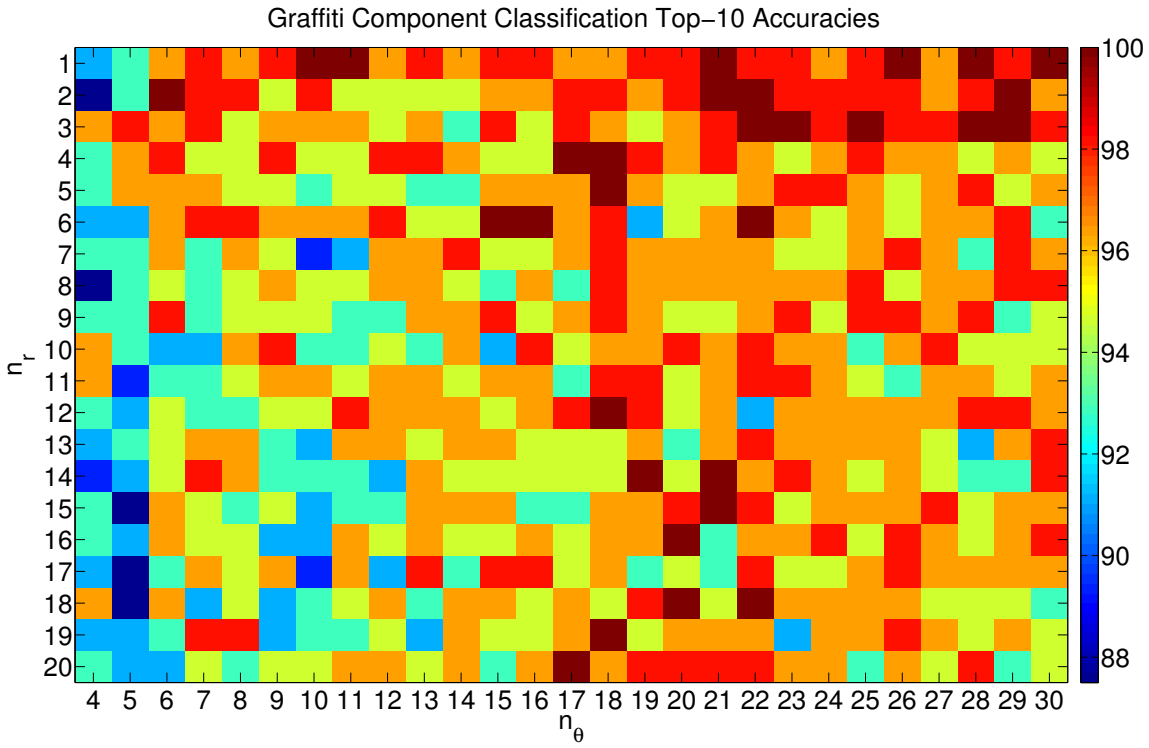


Fig. 5.17.: Color Map of Top-10 Classification Accuracies of Gang Graffiti Component Classification Using Different Values of n_r and n_θ.

Table 5.11: Top-5 Classification Accuracies of Gang Graffiti Component Classification for $n_r \in [1 \ldots 20]$ and $n_\theta \in [4 \ldots 17]$ (percentage).

n_r \ n_θ	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**
1	76.79	87.50	89.29	92.86	96.43	92.86	96.43	92.86	92.86	91.07	89.29	92.86	96.43	92.86
2	78.57	91.07	94.64	91.07	94.64	92.86	94.64	89.29	89.29	92.86	92.86	94.64	96.43	96.43
3	83.93	89.29	94.64	89.29	87.50	91.07	94.64	92.86	94.64	91.07	91.07	96.43	92.86	94.64
4	91.07	87.50	96.43	92.86	91.07	89.29	92.86	92.86	92.86	92.86	92.86	92.86	94.64	96.43
5	89.29	89.29	92.86	91.07	91.07	91.07	89.29	94.64	87.50	91.07	92.86	92.86	91.07	92.86
6	85.71	89.29	92.86	98.21	92.86	92.86	89.29	89.29	91.07	92.86	92.86	96.43	96.43	91.07
7	83.93	85.71	89.29	89.29	91.07	92.86	89.29	89.29	91.07	92.86	91.07	92.86	91.07	89.29
8	82.14	87.50	92.86	89.29	89.29	92.86	91.07	94.64	89.29	91.07	87.50	92.86	92.86	91.07
9	87.50	89.29	94.64	87.50	89.29	89.29	85.71	92.86	91.07	92.86	91.07	91.07	91.07	92.86
10	92.86	89.29	91.07	85.71	85.71	94.64	89.29	92.86	89.29	89.29	94.64	87.50	91.07	91.07
11	89.29	85.71	87.50	91.07	91.07	92.86	89.29	87.50	94.64	92.86	92.86	92.86	96.43	87.50
12	87.50	83.93	92.86	87.50	92.86	89.29	91.07	89.29	92.86	94.64	89.29	92.86	87.50	94.64
13	83.93	85.71	91.07	89.29	92.86	85.71	89.29	92.86	91.07	89.29	91.07	89.29	94.64	89.29
14	85.71	83.93	87.50	92.86	92.86	89.29	92.86	91.07	91.07	91.07	92.86	91.07	94.64	92.86
15	89.29	82.14	87.50	91.07	89.29	91.07	87.50	91.07	89.29	89.29	91.07	94.64	87.50	87.50
16	89.29	83.93	91.07	94.64	91.07	89.29	87.50	92.86	89.29	94.64	92.86	89.29	94.64	91.07
17	85.71	83.93	87.50	91.07	92.86	87.50	85.71	87.50	89.29	91.07	92.86	94.64	96.43	91.07
18	89.29	80.36	94.64	91.07	92.86	89.29	89.29	91.07	89.29	92.86	91.07	91.07	92.86	92.86
19	85.71	89.29	89.29	89.29	92.86	89.29	85.71	89.29	85.71	89.29	94.64	91.07	89.29	91.07
20	83.93	85.71	83.93	91.07	91.07	87.50	89.29	89.29	92.86	92.86	92.86	89.29	92.86	91.07

Table 5.12: Top-5 Classification Accuracies of Gang Graffiti Component Classification for $n_r \in [1 \ldots 20]$ and $n_\theta \in [18 \ldots 30]$ (percentage).

n_r \ n_θ	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**
1	96.43	98.21	98.21	100.00	98.21	98.21	96.43	98.21	100.00	96.43	100.00	98.21	100.00
2	98.21	96.43	98.21	100.00	100.00	98.21	98.21	98.21	98.21	96.43	98.21	100.00	96.43
3	96.43	94.64	96.43	98.21	100.00	100.00	98.21	100.00	98.21	98.21	100.00	100.00	98.21
4	100.00	98.21	96.43	98.21	96.43	94.64	96.43	98.21	96.43	96.43	94.64	96.43	94.64
5	100.00	96.43	94.64	94.64	96.43	98.21	98.21	96.43	94.64	96.43	98.21	94.64	96.43
6	98.21	91.07	94.64	96.43	100.00	96.43	94.64	96.43	94.64	96.43	96.43	98.21	92.86
7	98.21	96.43	96.43	96.43	96.43	94.64	94.64	96.43	98.21	96.43	92.86	98.21	96.43
8	98.21	96.43	96.43	96.43	96.43	96.43	96.43	98.21	94.64	96.43	96.43	98.21	98.21
9	98.21	96.43	94.64	94.64	96.43	98.21	94.64	98.21	98.21	96.43	98.21	92.86	94.64
10	96.43	96.43	98.21	96.43	98.21	96.43	96.43	92.86	96.43	98.21	94.64	94.64	94.64
11	98.21	98.21	94.64	96.43	98.21	98.21	96.43	94.64	92.86	96.43	96.43	94.64	96.43
12	100.00	98.21	94.64	96.43	91.07	96.43	96.43	96.43	96.43	96.43	98.21	98.21	96.43
13	94.64	96.43	92.86	96.43	98.21	96.43	96.43	96.43	96.43	94.64	91.07	96.43	98.21
14	94.64	100.00	94.64	100.00	96.43	98.21	96.43	94.64	96.43	94.64	92.86	92.86	98.21
15	96.43	96.43	98.21	100.00	98.21	94.64	96.43	96.43	96.43	98.21	94.64	96.43	96.43
16	96.43	96.43	100.00	92.86	96.43	96.43	98.21	94.64	98.21	96.43	94.64	96.43	98.21
17	96.43	92.86	94.64	92.86	98.21	94.64	94.64	96.43	98.21	96.43	96.43	96.43	96.43
18	94.64	98.21	100.00	94.64	100.00	96.43	96.43	96.43	96.43	94.64	94.64	94.64	92.86
19	100.00	94.64	96.43	96.43	96.43	91.07	96.43	96.43	98.21	96.43	94.64	96.43	94.64
20	96.43	98.21	98.21	98.21	98.21	96.43	96.43	92.86	96.43	94.64	98.21	92.86	94.64

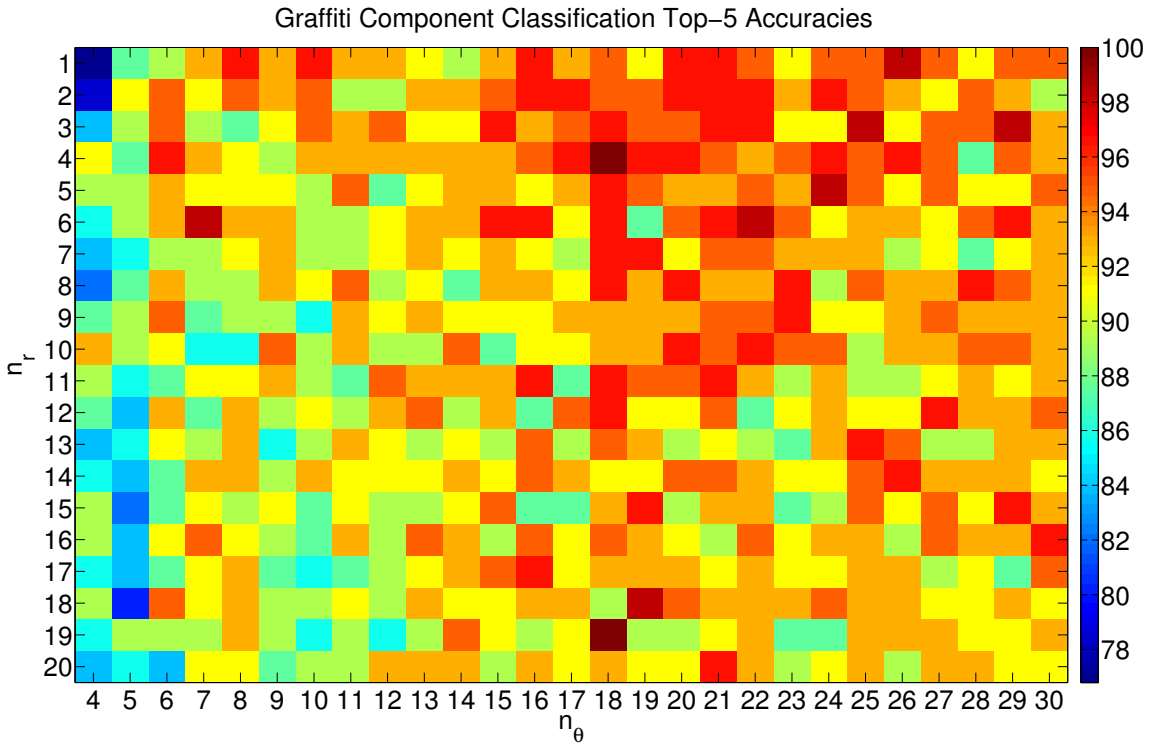


Fig. 5.18.: Color Map of Top-5 Classification Accuracies of Gang Graffiti Component Classification Using Different Values of n_r and n_θ.

Table 5.13: Classification Accuracy, Precision, Recall and F_1 Score for Each Class.

Class	**Accuracy**	**Precision**	**Recall**	F_1 **Score**
0	100%	100%	100%	1
8	100%	66.67%	100%	0.80
G	75%	100%	75%	0.86
3	100%	100%	100%	1
E	75%	100%	75%	0.86
s	50%	100%	50%	0.67
2	100%	80%	100%	0.89
1	100%	80%	100%	0.89
x	100%	100%	100%	1
5-point star	100%	80%	100%	0.89
6-point star	100%	100%	100%	1
4	75%	100%	75%	0.86
pitchfork	100%	80%	100%	0.89
arrow	75%	100%	75%	0.86

corresponds to predicted outputs and the y-axis corresponds to expected outputs, precision P_i and recall R_i for class i are defined as

$$P_i = \frac{M_{ii}}{\sum_j M_{ji}} \tag{5.8}$$

$$R_i = \frac{M_{ii}}{\sum_j M_{ij}}. \tag{5.9}$$

Given precision and recall values, the F_{1i} score is given by

$$F_{1i} = 2\frac{P_i R_i}{P_i + R_i} \tag{5.10}$$

As a comparison, Tables 5.14 to 5.16 show the classification accuracies, Top-10 accuracies and Top-5 accuracies when using SIFT descriptors instead of LSC descriptors. The maximum classification accuracy achieved is 41.07% with $n_r = 6$ and $n_\theta = 13$, with a Top-10 accuracy of 75.00% and a Top-5 accuracy of 55.36%. The

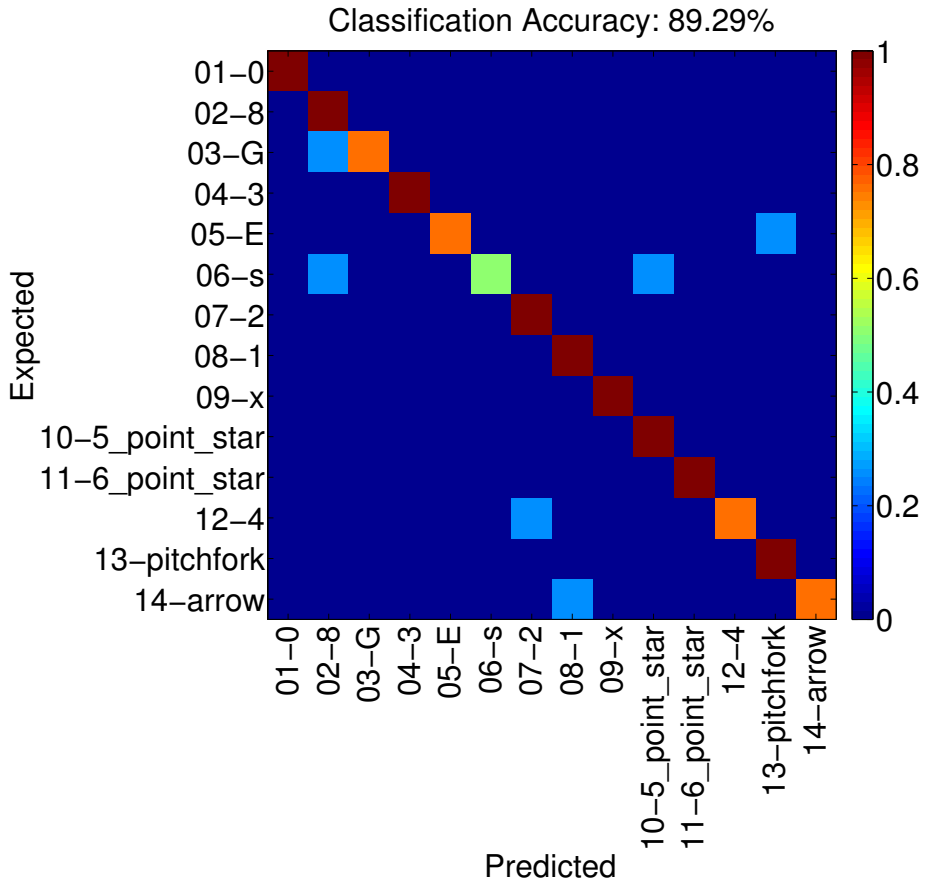


Fig. 5.19.: Confusion Matrix for the 14 Graffiti Component Classes.

average query time is the same as when using LSC descriptors, because most of the time is spend pushing the descriptors down the vocabulary tree.

In order to evaluate the overall performance of our "Gang Graffiti Component Classification" system we also used the Mean Average Precision (MAP) measure, which provides a single-figure measure of quality across recall levels and has been shown to have especially good discrimination and stability [308–310].

The MAP is defined as

$$MAP = \frac{1}{Q} \sum_{j=1}^{Q} \frac{1}{N} \sum_{k=1}^{N} P_{jk}, \tag{5.11}$$

where Q is the total number of query images and N is the number of database images retrieved for each query. Equation 5.11 can be redefined as the average precision scores for the set of queries:

$$MAP = \frac{\sum_{j=1}^{Q} AveP(j)}{Q}, \tag{5.12}$$

where $AveP(j)$ is average precision of the j-th query image, defined as

$$AveP(j) = \frac{\sum_{k=1}^{N} P_{jk}}{N}, \tag{5.13}$$

being P_{jk} is the precision of the j-th query image at rank k:

$$P_{jk} = \frac{\sum_{i=1}^{k} I_{ji}}{k}. \tag{5.14}$$

Table 5.14: Classification Accuracies of Gang Graffiti Component Classification for $n_r \in [1 \dots 10]$ and $n_\theta \in [4 \dots 17]$ using SIFT Descriptors (percentage).

n_r \ n_θ	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**
1	33.93	35.71	32.14	37.50	33.93	37.50	32.14	30.36	30.36	32.14	39.29	35.71	30.36	33.93
2	33.93	28.57	32.14	32.14	30.36	32.14	33.93	37.50	28.57	30.36	30.36	33.93	30.36	30.36
3	32.14	30.36	30.36	28.57	33.93	35.71	28.57	35.71	33.93	32.14	33.93	30.36	33.93	32.14
4	26.79	33.93	32.14	37.50	41.07	35.71	30.36	33.93	33.93	33.93	33.93	30.36	35.71	30.36
5	30.36	32.14	33.93	33.93	35.71	30.36	28.57	33.93	30.36	30.36	35.71	32.14	32.14	28.57
6	28.57	30.36	30.36	33.93	33.93	33.93	30.36	32.14	37.50	41.07	35.71	28.57	33.93	33.93
7	32.14	32.14	32.14	35.71	35.71	32.14	30.36	39.29	33.93	32.14	35.71	30.36	25.00	32.14
8	32.14	28.57	33.93	30.36	32.14	33.93	26.79	32.14	33.93	30.36	35.71	35.71	30.36	33.93
9	32.14	35.71	32.14	32.14	28.57	30.36	35.71	30.36	33.93	30.36	35.71	35.71	33.93	33.93
10	30.36	32.14	35.71	33.93	32.14	33.93	33.93	33.93	30.36	33.93	35.71	30.36	32.14	30.36

Table 5.15: Top-10 Classification Accuracies of Gang Graffiti Component Classification for $n_r \in [1 \ldots 10]$ and $n_\theta \in [4 \ldots 17]$ using SIFT Descriptors (percentage).

n_r \ n_θ	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**
1	73.21	76.79	75.00	69.64	64.29	67.86	67.86	69.64	71.43	67.86	73.21	67.86	66.07	69.64
2	73.21	69.64	75.00	71.43	67.86	73.21	71.43	71.43	67.86	67.86	66.07	64.29	73.21	73.21
3	69.64	75.00	78.57	71.43	67.86	69.64	69.64	76.79	64.29	67.86	67.86	76.79	66.07	71.43
4	69.64	71.43	66.07	69.64	66.07	75.00	66.07	67.86	64.29	67.86	76.79	69.64	78.57	73.21
5	73.21	71.43	73.21	71.43	67.86	67.86	75.00	69.64	78.57	71.43	73.21	69.64	69.64	66.07
6	73.21	75.00	69.64	69.64	73.21	62.50	73.21	75.00	66.07	75.00	69.64	69.64	69.64	67.86
7	73.21	75.00	71.43	67.86	75.00	75.00	62.50	75.00	67.86	69.64	69.64	69.64	71.43	69.64
8	71.43	78.57	69.64	66.07	75.00	64.29	75.00	71.43	71.43	69.64	71.43	69.64	76.79	69.64
9	62.50	66.07	67.86	66.07	67.86	80.36	69.64	60.71	78.57	69.64	71.43	71.43	75.00	67.86
10	71.43	64.29	67.86	73.21	62.50	71.43	71.43	69.64	66.07	76.79	66.07	69.64	67.86	67.86

Table 5.16: Top-5 Classification Accuracies of Gang Graffiti Component Classification for $n_r \in [1 \dots 10]$ and $n_\theta \in [4 \dots 17]$ using SIFT Descriptors (percentage).

n_r \ n_θ	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**
1	51.79	50.00	57.14	53.57	48.21	53.57	55.36	53.57	53.57	55.36	64.29	53.57	53.57	48.21
2	60.71	50.00	58.93	50.00	53.57	53.57	55.36	55.36	51.79	53.57	51.79	50.00	53.57	50.00
3	55.36	48.21	58.93	55.36	57.14	55.36	51.79	58.93	48.21	53.57	55.36	62.50	53.57	58.93
4	50.00	58.93	48.21	55.36	55.36	58.93	50.00	58.93	48.21	50.00	55.36	55.36	67.86	53.57
5	55.36	57.14	62.50	57.14	53.57	57.14	53.57	53.57	55.36	57.14	57.14	51.79	58.93	53.57
6	58.93	55.36	53.57	55.36	53.57	53.57	57.14	53.57	55.36	55.36	55.36	50.00	62.50	48.21
7	58.93	51.79	53.57	57.14	66.07	57.14	51.79	62.50	57.14	53.57	58.93	58.93	50.00	53.57
8	51.79	55.36	55.36	53.57	51.79	50.00	51.79	55.36	46.43	53.57	55.36	50.00	55.36	55.36
9	51.79	55.36	53.57	57.14	50.00	55.36	58.93	51.79	53.57	55.36	51.79	51.79	60.71	53.57
10	53.57	50.00	58.93	55.36	50.00	57.14	57.14	60.71	51.79	57.14	51.79	57.14	57.14	55.36

Table 5.17: Example of MAP score calculation for a set of two queries. The total MAP score is $\frac{0.22+0.41}{2} = 0.31$.

Prediction	**Correctness**	**Precision**
1	wrong	none
2	right	1/2
3	right	2/3
4	wrong	none
5	right	3/5
6	wrong	none
7	wrong	none
8	wrong	none
9	right	4/9
10	wrong	none

(a) $AveP = \frac{1/2+2/3+3/5+4/9}{10} = 0.22$

Prediction	**Correctness**	**Precision**
1	right	1/1
2	right	2/2
3	right	3/3
4	wrong	none
5	wrong	none
6	wrong	none
7	wrong	none
8	right	4/8
9	right	5/9
10	wrong	none

(b) $AveP = \frac{1/1+2/2+3/3+4/8+5/9}{10} = 0.41$

I_{ji} is an indicator function equaling 1 if the j-th query image at rank k is a match, and zero otherwise. Table 5.17 shows an example of how to calculate the MAP score with $Q = 2$ and $N = 10$. In our experiments $Q = 56$ and $N = 10$.

Tables 5.18 and 5.19 show the MAP scores for a range of n_r and n_θ using LSC descriptors. Figure 5.20 illustrates the same information using a color map. This results confirm that not using enough bins for radius and angles, or using too many, will cause the classification accuracy to drop. Values of $n_r \in [3 \dots 18]$ and $n_\theta \in$

[12 . . . 28] provide enough discrimination between feature locations and robustness against shape elasticity.

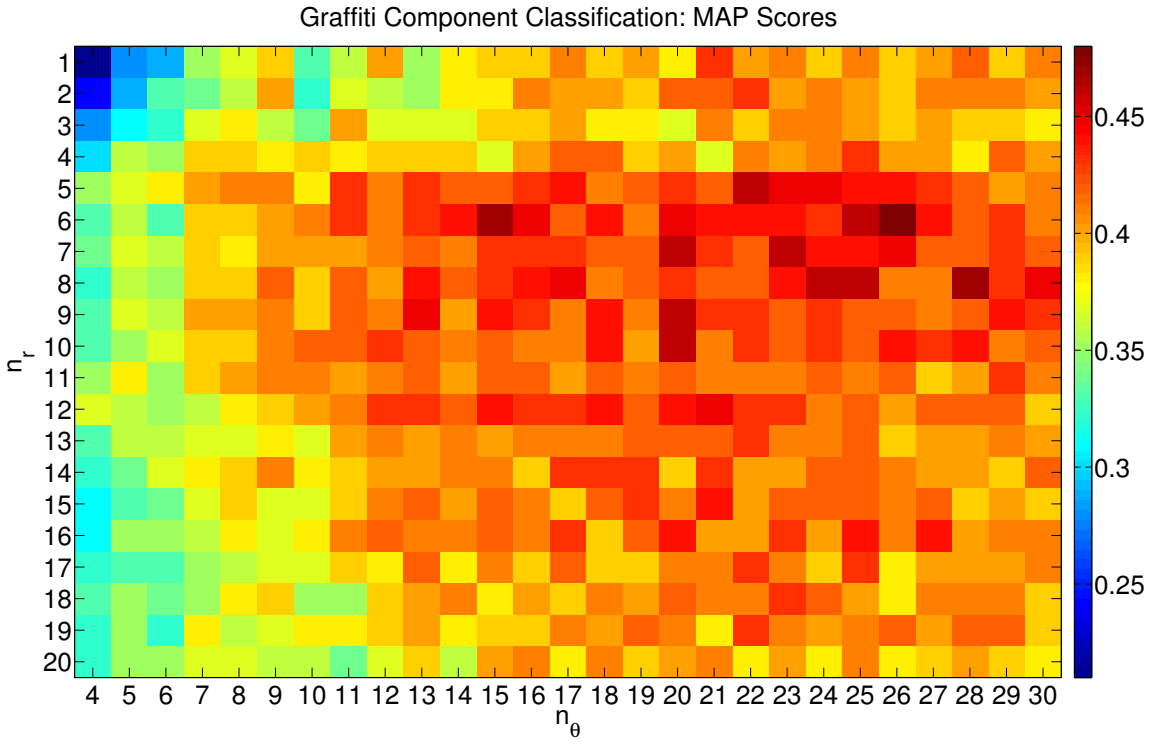


Fig. 5.20.: Color Map of MAP Scores of Gang Graffiti Component Classification Using Different Values of n_r and n_θ.

5.1.4 End-To-End System

In this experiment we tested the entire GARI system, including all the steps shown in Figure 5.21. The system is composed of seven blocks: Color Recognition Based on Touchscreen Tracing, Color Correction Based on Mobile Light Sensor, Color Image Segmentation Based on Gaussian Thresholding, Block-Wise Gaussian Segmentation Enhancement, Background Stripe Removal, Graffiti Component Reconnection, and Graffiti Component Classification. Note that the Color Recognition Based on Touch-

Table 5.18: MAP Scores of Gang Graffiti Component Classification for $n_r \in [1 \ldots 20]$ and $n_\theta \in [4 \ldots 17]$ (percentage).

n_r \ n_θ	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**
1	0.23	0.28	0.32	0.37	0.40	0.37	0.35	0.37	0.39	0.39	0.39	0.39	0.41	0.42
2	0.23	0.27	0.33	0.35	0.37	0.39	0.34	0.35	0.38	0.36	0.39	0.42	0.41	0.40
3	0.32	0.34	0.34	0.39	0.38	0.39	0.37	0.37	0.38	0.35	0.37	0.38	0.38	0.39
4	0.34	0.39	0.39	0.41	0.40	0.39	0.41	0.41	0.39	0.40	0.41	0.42	0.40	0.42
5	0.35	0.39	0.40	0.42	0.40	0.41	0.41	0.44	0.43	0.41	0.42	0.45	0.44	0.44
6	0.34	0.39	0.38	0.41	0.41	0.41	0.39	0.41	0.42	0.44	0.43	0.43	0.43	0.46
7	0.36	0.37	0.36	0.40	0.40	0.40	0.39	0.42	0.41	0.41	0.43	0.44	0.44	0.42
8	0.36	0.37	0.37	0.39	0.39	0.42	0.41	0.41	0.41	0.44	0.43	0.42	0.41	0.43
9	0.34	0.39	0.37	0.41	0.41	0.42	0.39	0.42	0.39	0.42	0.43	0.44	0.44	0.43
10	0.34	0.39	0.39	0.41	0.40	0.43	0.40	0.42	0.41	0.43	0.43	0.42	0.43	0.42
11	0.36	0.39	0.39	0.40	0.42	0.42	0.43	0.42	0.42	0.43	0.44	0.46	0.44	0.44
12	0.35	0.36	0.38	0.41	0.43	0.41	0.41	0.39	0.42	0.43	0.42	0.42	0.42	0.42
13	0.34	0.36	0.38	0.43	0.39	0.38	0.38	0.42	0.40	0.42	0.41	0.43	0.41	0.42
14	0.36	0.37	0.36	0.40	0.41	0.40	0.41	0.41	0.40	0.40	0.41	0.39	0.42	0.41
15	0.34	0.36	0.36	0.41	0.42	0.42	0.39	0.41	0.42	0.44	0.43	0.43	0.41	0.41
16	0.34	0.37	0.37	0.40	0.40	0.41	0.42	0.41	0.43	0.41	0.42	0.42	0.42	0.43
17	0.34	0.34	0.37	0.42	0.40	0.38	0.41	0.41	0.40	0.43	0.41	0.41	0.42	0.42
18	0.35	0.36	0.37	0.39	0.39	0.39	0.41	0.42	0.42	0.42	0.39	0.44	0.42	0.43
19	0.34	0.36	0.40	0.40	0.38	0.40	0.38	0.39	0.42	0.41	0.42	0.39	0.42	0.41
20	0.35	0.36	0.37	0.39	0.39	0.41	0.39	0.40	0.41	0.41	0.39	0.40	0.42	0.42

Table 5.19: MAP Scores of Gang Graffiti Component Classification for $n_r \in [1 \dots 20]$ and $n_\theta \in [18 \dots 30]$ (percentage).

n_r \ n_θ	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**
1	0.40	0.42	0.39	0.41	0.43	0.42	0.41	0.41	0.42	0.42	0.39	0.40	0.42
2	0.40	0.42	0.39	0.42	0.41	0.41	0.41	0.40	0.40	0.41	0.43	0.41	0.43
3	0.40	0.42	0.39	0.40	0.41	0.39	0.40	0.40	0.41	0.40	0.40	0.41	0.41
4	0.41	0.38	0.42	0.43	0.41	0.42	0.41	0.41	0.41	0.42	0.40	0.42	0.42
5	0.43	0.44	0.43	0.44	0.44	0.45	0.43	0.45	0.44	0.44	0.44	0.45	0.45
6	0.44	0.44	0.46	0.42	0.46	0.46	0.47	0.45	0.45	0.45	0.46	0.44	0.43
7	0.44	0.44	0.43	0.46	0.44	0.45	0.45	0.45	0.46	0.44	0.44	0.44	0.43
8	0.43	0.43	0.43	0.43	0.43	0.43	0.45	0.43	0.44	0.44	0.43	0.45	0.44
9	0.43	0.44	0.47	0.43	0.44	0.42	0.44	0.44	0.43	0.43	0.44	0.45	0.45
10	0.44	0.43	0.44	0.43	0.43	0.43	0.43	0.42	0.44	0.42	0.42	0.45	0.43
11	0.42	0.43	0.44	0.44	0.45	0.44	0.42	0.43	0.44	0.42	0.41	0.45	0.43
12	0.43	0.44	0.44	0.42	0.44	0.44	0.42	0.43	0.43	0.43	0.42	0.47	0.43
13	0.43	0.41	0.44	0.44	0.43	0.44	0.42	0.43	0.42	0.42	0.41	0.42	0.41
14	0.42	0.42	0.44	0.43	0.43	0.42	0.43	0.43	0.42	0.40	0.40	0.42	0.42
15	0.42	0.42	0.44	0.44	0.42	0.44	0.42	0.44	0.42	0.43	0.43	0.41	0.44
16	0.43	0.43	0.42	0.43	0.40	0.45	0.44	0.43	0.41	0.43	0.43	0.41	0.39
17	0.43	0.43	0.45	0.41	0.43	0.43	0.42	0.44	0.41	0.44	0.41	0.43	0.40
18	0.42	0.41	0.43	0.43	0.45	0.43	0.43	0.42	0.42	0.43	0.42	0.42	0.39
19	0.43	0.42	0.43	0.42	0.43	0.44	0.44	0.42	0.41	0.43	0.42	0.42	0.40
20	0.43	0.40	0.42	0.43	0.43	0.41	0.42	0.41	0.42	0.41	0.41	0.42	0.41

screen Tracing is the only step that is done on the mobile device. The rest of the process is done in the server.

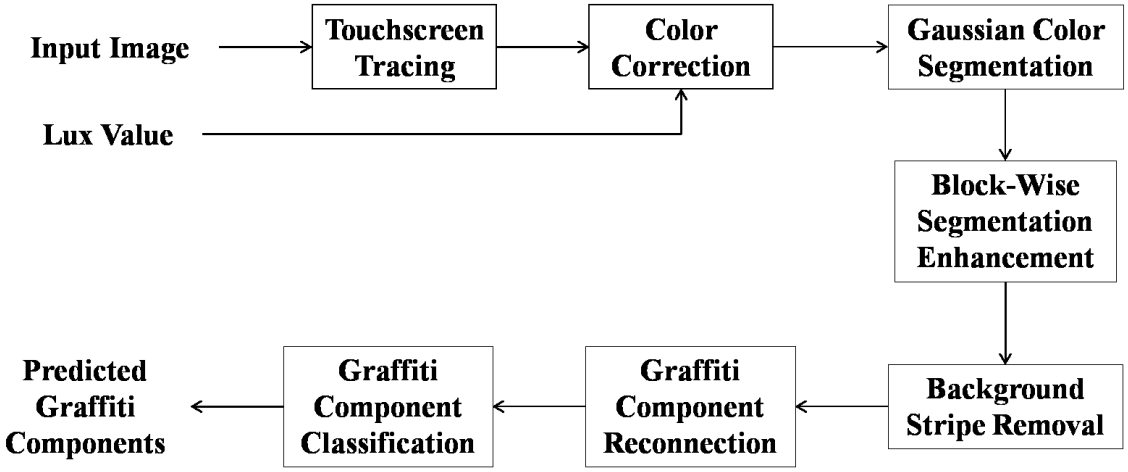


Fig. 5.21.: GARI End-To-End System.

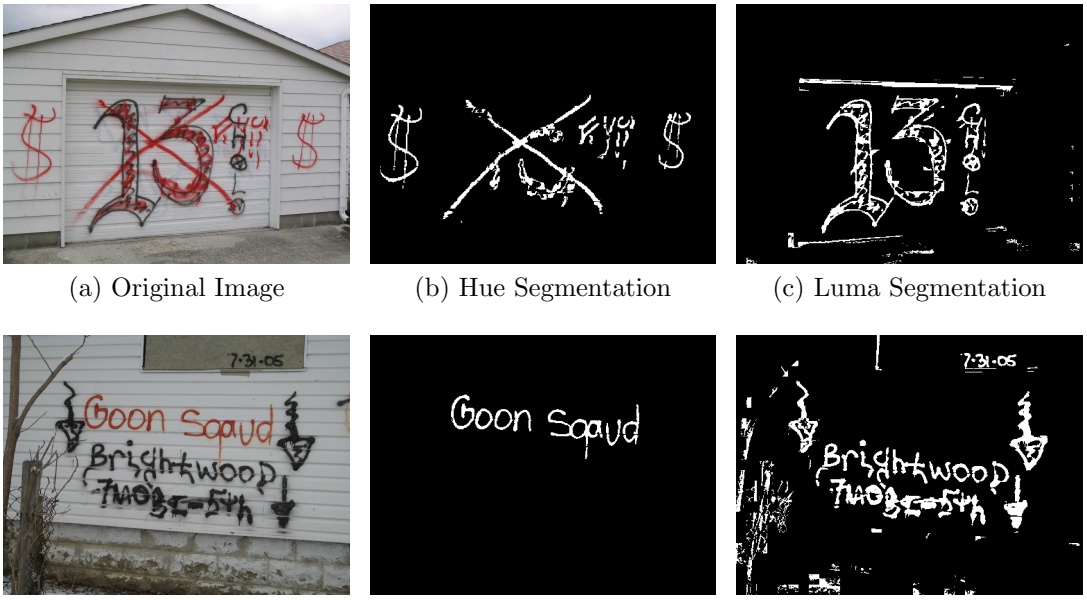
We use the touchscreen tracing method to obtain the color median (either luma or hue) of a graffiti component, and we send this information to the server along with the image and the lux value automatically obtained from the device's light sensor. Once on the server we color correct the image by mapping the lux value to a color correction matrix. We then use the color median to automatically segment the image using our proposed Gaussian thresholding method. The segmented image is locally enhanced, the existing background stripes are removed and the disjoint connected components are reconnected. The extracted components are gang graffiti component candidates that are classified and the predicted results are returned the mobile device.

We tested the entire process in 20 images with different colors, shapes, backgrounds, lighting conditions, and taken in different seasons (Summer and Winter). Figure 5.22 illustrates the 20 images.

Table 5.20 shows the running times of each step for all the test images. The processing times vary from 3.15 to 10.39 seconds, with a median of 4.69 seconds. Images 1016 and 1019 have two versions each because we segmented them using

Fig. 5.22.: Test Images for Automatic Gang Graffiti Segmentation.

different colors (i.e. two different touchscreen tracings). Figure 5.23 illustrates these cases. Figure 5.21 shows the running times of the three main blocks: color correction, image segmentation, and component classification. The high standard deviations of some steps indicate their dependence of the complexity of the input image. For example, the Graffiti Component Reconnection step evaluates each end-point of the image skeleton. The more complex the graffiti is, the more end-points it will have, and the more time it will take to process. Also, depending on the graffiti the number of segmented components vary from 4 to 31. This affects the running time of the Content Based Image Retrieval method following the segmentation process.



(a) Original Image (b) Hue Segmentation (c) Luma Segmentation

(d) Original Image (e) Hue Segmentation (f) Luma Segmentation

Fig. 5.23.: Images Segmented Separately From Two Different TouchScreen Tracings.

Figure 5.24 shows some examples of the proposed Color Image Segmentation Based on Gaussian Thresholding followed by Block-Wise Gaussian Segmentation Enhancement. Note that the enhancement contributes to both noise removal and graffiti component reconstruction. Figure C.22 shows some examples of our color image segmentation compared against other thresholding methods, including Niblack [20] (local thresholding) and Otsu [294] (global thresholding). For Niblack we set the

Table 5.20: Running Times (seconds) of Each Step in The GARI End-To-End System. 1: Color Correction Based on Mobile Light Sensor, 2: Color Image Segmentation Based on Gaussian Thresholding, 3: Block-Wise Gaussian Segmentation Enhancement, 4: Background Stripe Removal, 5: Graffiti Component Reconnection, 6: Graffiti Component Classification.

Image Number	**1**	**2**	**3**	**4**	**5**	**6**	**Total**
1001	1.72	0.24	0.88	0.05	2.05	0.64	5.57
1002	1.91	0.13	0.52	0.24	3.26	1.07	7.12
1003	1.85	0.28	0.94	0.05	0.91	0.99	5.04
1004	1.69	0.60	1.28	0.04	0.35	0.64	4.60
1005	2.27	0.17	0.79	0.03	0.28	0.64	4.18
1006	2.05	0.12	0.47	0.04	0.76	0.64	4.08
1007	1.71	0.13	0.65	0.04	0.34	0.28	3.15
1008	1.69	0.12	0.46	0.04	0.74	0.71	3.76
1009	1.73	0.25	0.62	0.04	0.75	0.99	4.39
1010	1.75	0.61	1.03	0.19	3.01	2.20	8.79
1011	1.87	0.19	0.62	0.07	5.86	1.78	10.39
1012	1.92	0.78	1.23	0.07	2.89	0.92	7.81
1013	1.70	0.20	0.85	0.04	0.75	0.50	4.04
1014	1.73	0.73	1.21	0.04	0.57	0.50	4.77
1015	1.67	0.76	1.19	0.05	2.00	1.07	6.73
1016_1	1.84	0.20	0.89	0.05	0.97	0.57	4.51
1016_2	1.80	0.19	0.61	0.05	1.10	0.43	4.17
1017	2.30	0.15	0.66	0.04	1.05	0.85	5.05
1018	1.86	0.14	0.73	0.04	0.39	0.43	3.58
1019_1	1.92	0.56	1.24	0.03	0.15	1.78	5.68
1019_2	1.71	0.55	1.05	0.09	2.51	0.36	6.27
1020	1.76	0.13	0.56	0.05	0.55	0.50	3.54
Median	**1.78**	**0.20**	**0.82**	**0.05**	**0.84**	**0.64**	**4.69**
Std Dev	0.17	0.24	0.27	0.05	1.38	0.50	1.85

Table 5.21: Running Times (seconds) of The Three Main Blocks in The GARI End-To-End System. 1: Color Correction, 2: Automatic Graffiti Component Segmentation, 3: Graffiti Component Classification. CCs: Number of Connected Components.

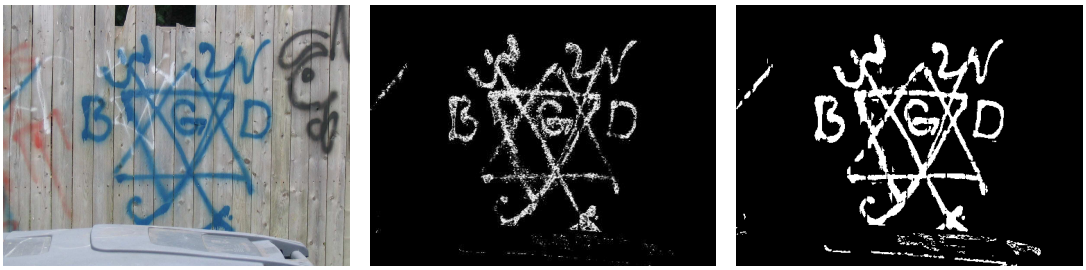
Image Number	**1**	**2**	**CCs**	**3**	**Total**
1001	1.72	3.22	9	0.64	5.57
1002	1.91	4.15	15	1.07	7.12
1003	1.85	2.19	14	0.99	5.04
1004	1.69	2.27	9	0.64	4.60
1005	2.27	1.27	9	0.64	4.18
1006	2.05	1.39	9	0.64	4.08
1007	1.71	1.16	4	0.28	3.15
1008	1.69	1.36	10	0.71	3.76
1009	1.73	1.66	14	0.99	4.39
1010	1.75	4.84	31	2.20	8.79
1011	1.87	6.74	25	1.78	10.39
1012	1.92	4.97	13	0.92	7.81
1013	1.70	1.84	7	0.50	4.04
1014	1.73	2.54	7	0.50	4.77
1015	1.67	4.00	15	1.07	6.73
1016_1	1.84	2.10	8	0.57	4.51
1016_2	1.80	1.95	6	0.43	4.17
1017	2.30	1.89	12	0.85	5.05
1018	1.86	1.29	6	0.43	3.58
1019_1	1.92	1.98	25	1.78	5.68
1019_2	1.71	4.20	5	0.36	6.27
1020	1.76	1.28	7	0.50	3.54
Median	**1.78**	**2.04**	**9**	**0.64**	**4.69**
Std Dev	0.17	1.52	7.04	0.50	1.85

filter radius to 25 pixels and standard deviation threshold to -0.2. When the graffiti surface has uniform texture and color all the methods produce good results. However, for complex surfaces and non-uniform illumination scenes Niblack and Otsu fail to segment the graffiti from the background. The only disadvantage of our proposed method is the running time. The average running times of Niblack and Otsu are 0.5 seconds and 0.01 seconds respectively, while our proposed method runs in 1 second on average. The comparison of the three methods for all 20 test images can be found on Appendix C. We also considered a stroke-width based image operator proposed in [311] to detect text in natural scenes, but it is not robust against non-alphanumerical symbols.

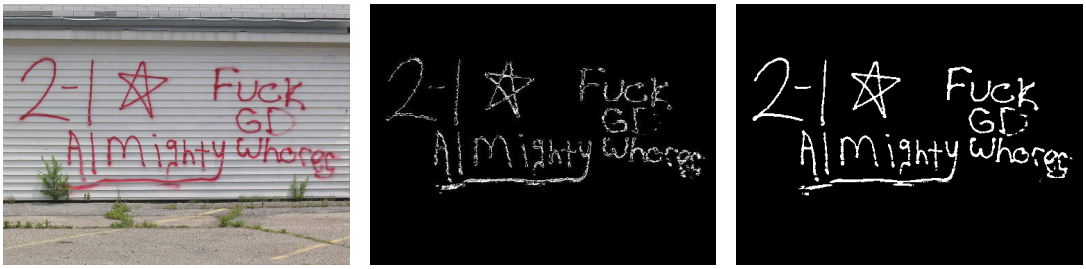
The Background Strip Removal process is the fastest of the four segmentation steps on average. This is because even though 18 of the 20 test images contain background strips only two of them still contain strips after the enhancement step. Figure 5.27 shows some examples of background strips removed during previous steps. Figure 5.26 shows the strip removal process in the two remaining images.

The Graffiti Component Reconnection process is the slowest of the four segmentation steps. This is because it conducts an exhaustive search among all the end-points on the image skeleton to find connection point candidates. Figure 5.28 shows an example of a test image where 252 end-points are checked in 5.86 seconds. Large amount of end-points are usually the results of skeletonization of background noise, such as trash on the ground or vegetation. Figure 5.29 shows some examples of successful component reconnection. Note that reconnection is not necessary when two end-points already belong to the same 8-neighbor connected component. Sometimes the distribution of the connected components is such that false connections are created, as shown in Figure 5.29b between the *1* in *2-1* and the *l* in *Almighty*.

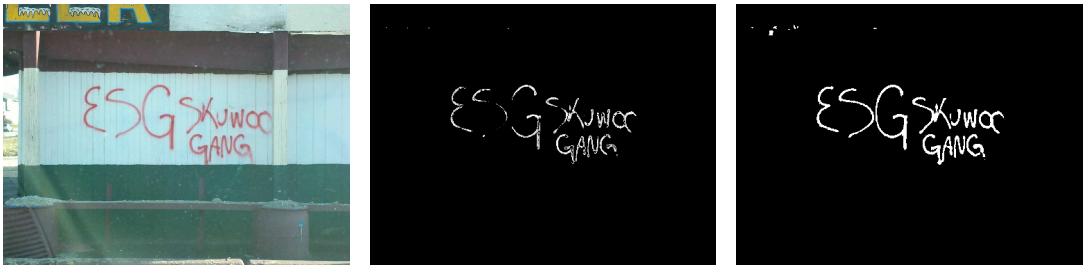
To illustrate the effectiveness of the automatic gang graffiti segmentation Figure 5.30 shows examples of the number of 8-neighbor connected components after Color Image Segmentation Based on Gaussian Thresholding, and after Graffiti Component Reconnection. An additional step can be added to merge connected components



(a) Original Image (b) Gaussian Thresholding (c) Block-Wise Enhancement



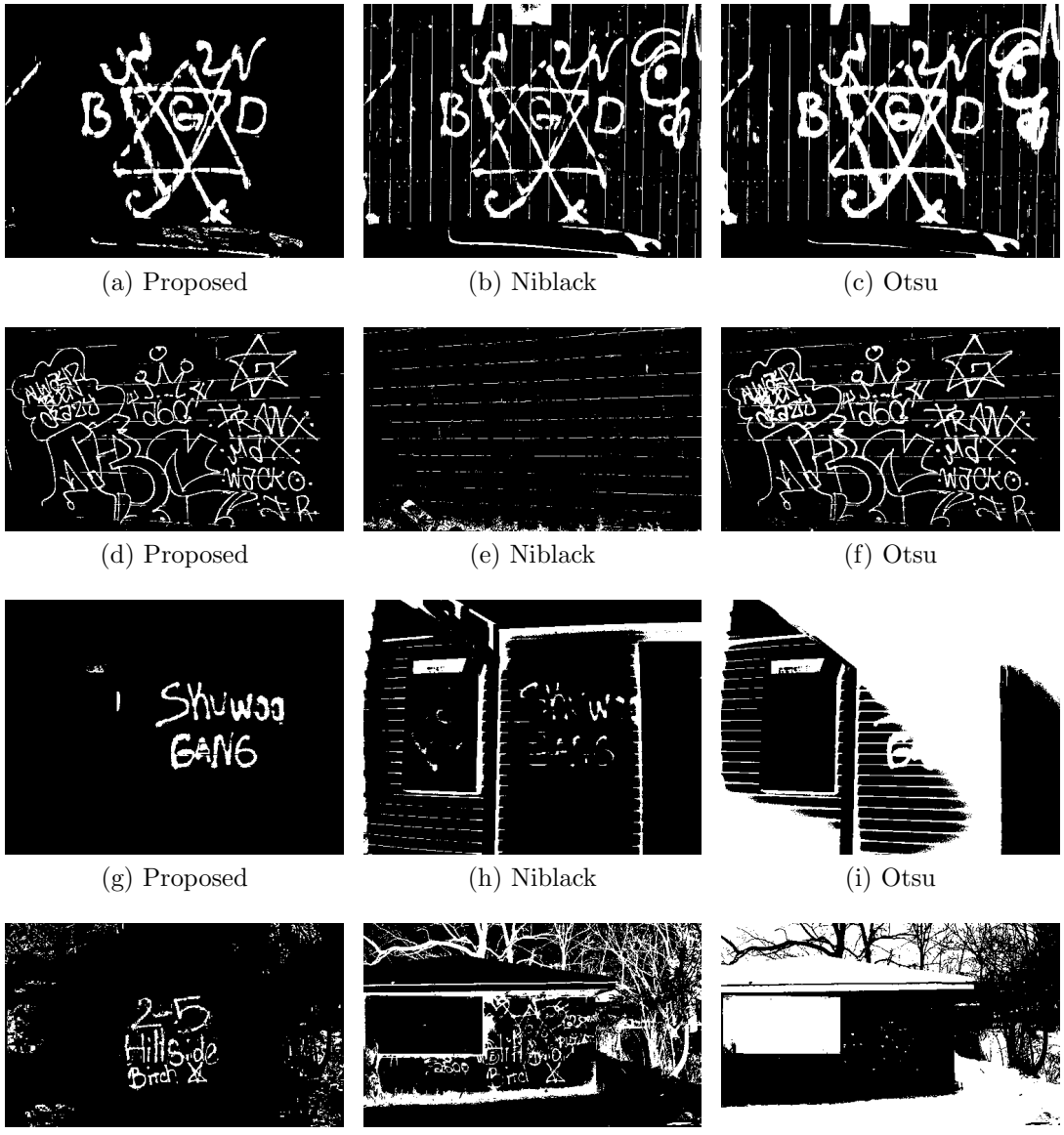
(d) Original Image (e) Gaussian Thresholding (f) Enhancement



(g) Original Image (h) Gaussian Thresholding (i) Enhancement

(j) Original Image (k) Gaussian Thresholding (l) Enhancement

Fig. 5.24.: Examples of our proposed Color Image Segmentation Based on Gaussian Thresholding followed by Block-Wise Gaussian Segmentation Enhancement.



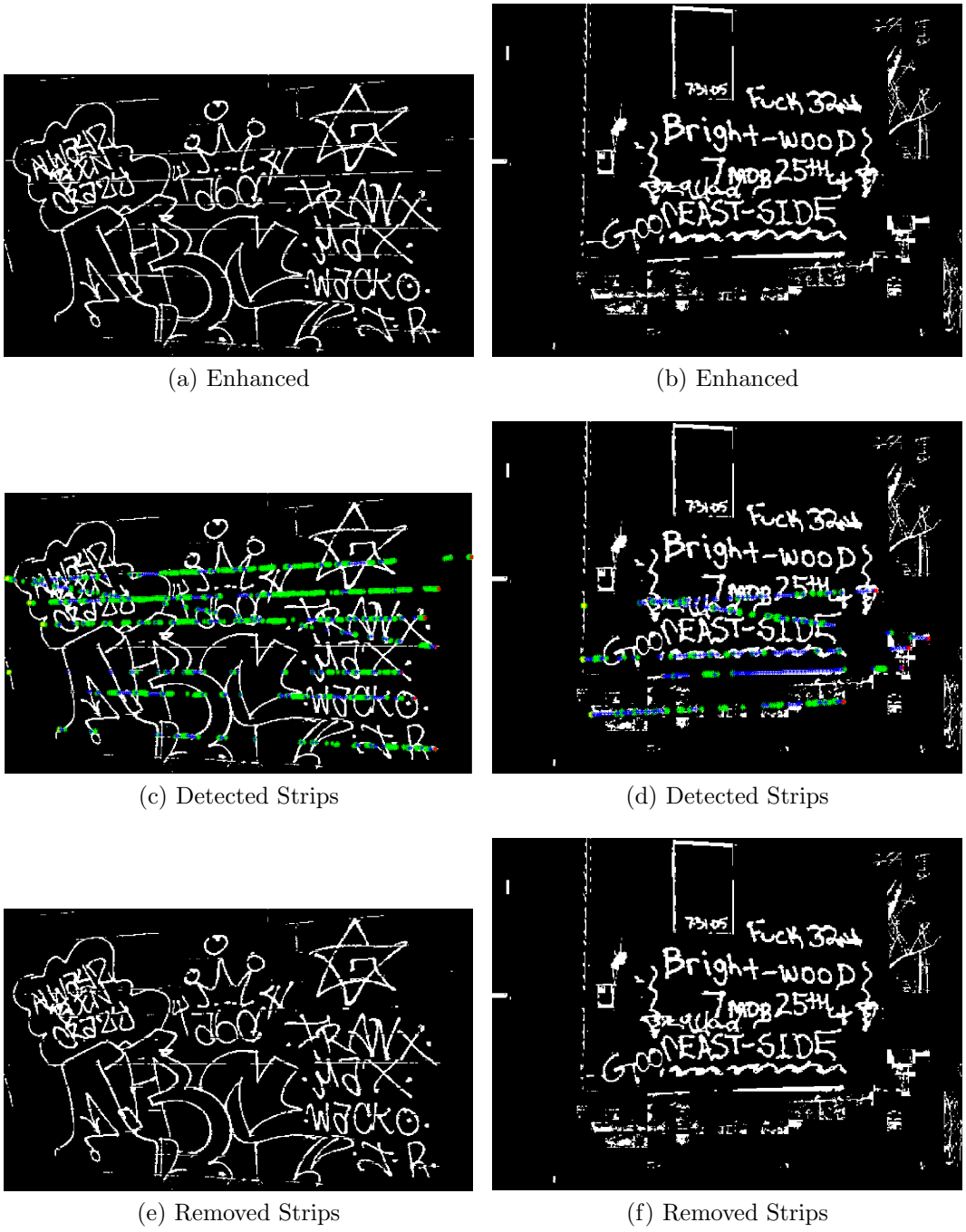
(a) Proposed (b) Niblack (c) Otsu

(d) Proposed (e) Niblack (f) Otsu

(g) Proposed (h) Niblack (i) Otsu

(j) Proposed (k) Niblack (l) Otsu

Fig. 5.25.: Comparison of our proposed color image segmentation method against Niblack and Otsu thresholding. From top to bottom: 1001, 1002, 1004, 1017.



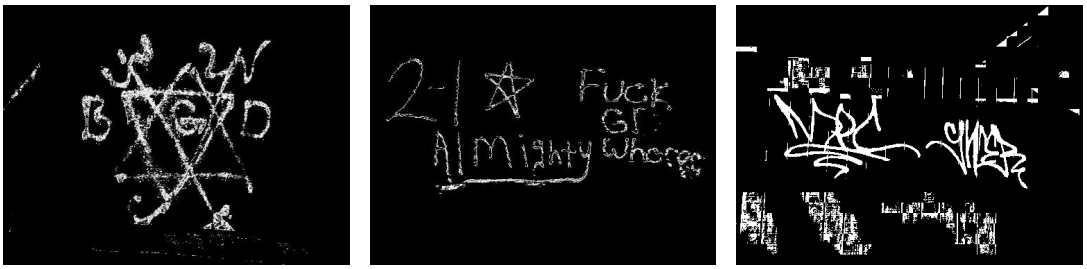
(a) Enhanced (b) Enhanced

(c) Detected Strips (d) Detected Strips

(e) Removed Strips (f) Removed Strips

Fig. 5.26.: Examples of Background Strip Removal.

(a) Original Image (b) Original Image (c) Original Image

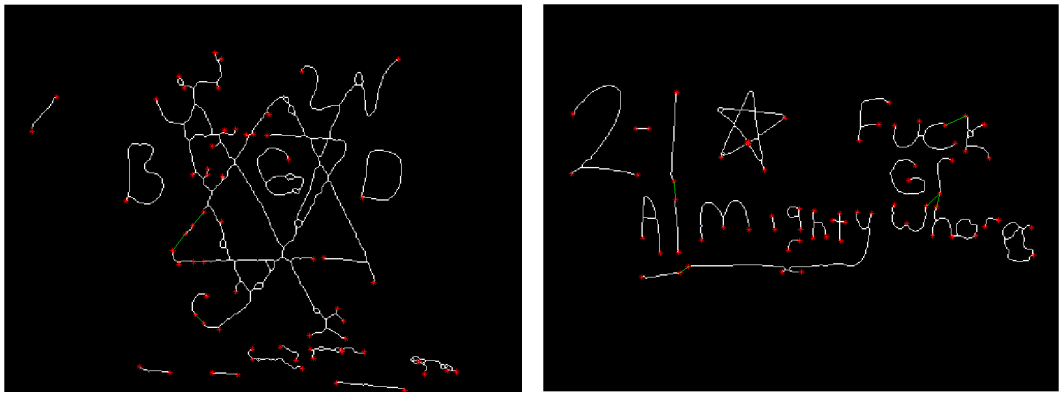


(d) Gaussian Thresholding (e) Gaussian Thresholding (f) Enhanced

Fig. 5.27.: Examples of Background Strips Automatically Removed in Previous Steps.

(a) Original Image (b) Skeleton End-Points

Fig. 5.28.: End-Points in Skeleton of Image 1011.



(a) 1001 - 4 Reconnections

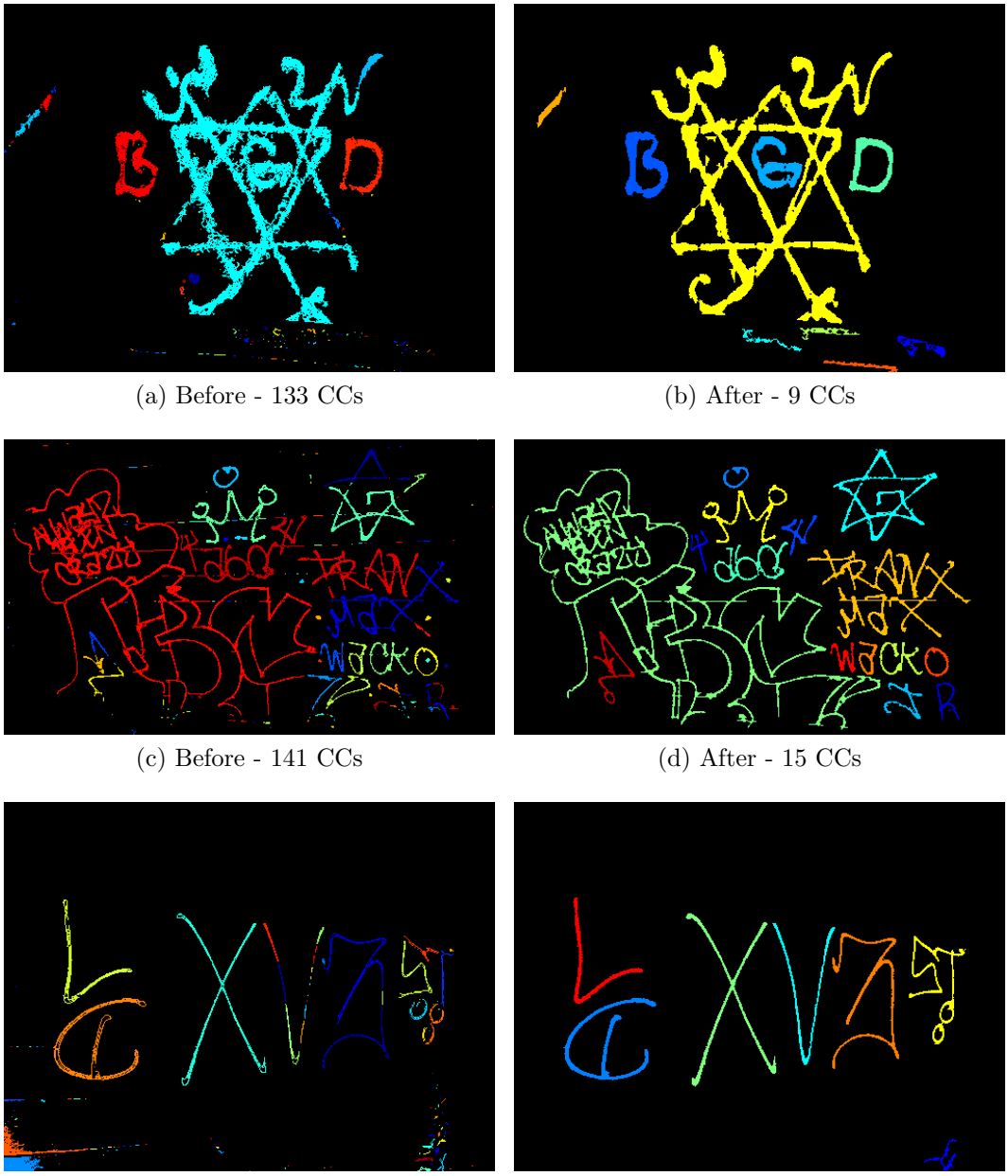
(b) 1003 - 5 Reconnections

(c) 1014 - 1 Reconnection

(d) 1020 - 1 Reconnection

Fig. 5.29.: Examples of Graffiti Component Reconnection.

that may belong together forming words, as shown in Figure 5.31. Note how graffiti components are successfully segmented and can be now treated separately for classification.



(a) Before - 133 CCs

(b) After - 9 CCs

(c) Before - 141 CCs

(d) After - 15 CCs

(e) Before - 114 CCs

(f) After - 7 CCs

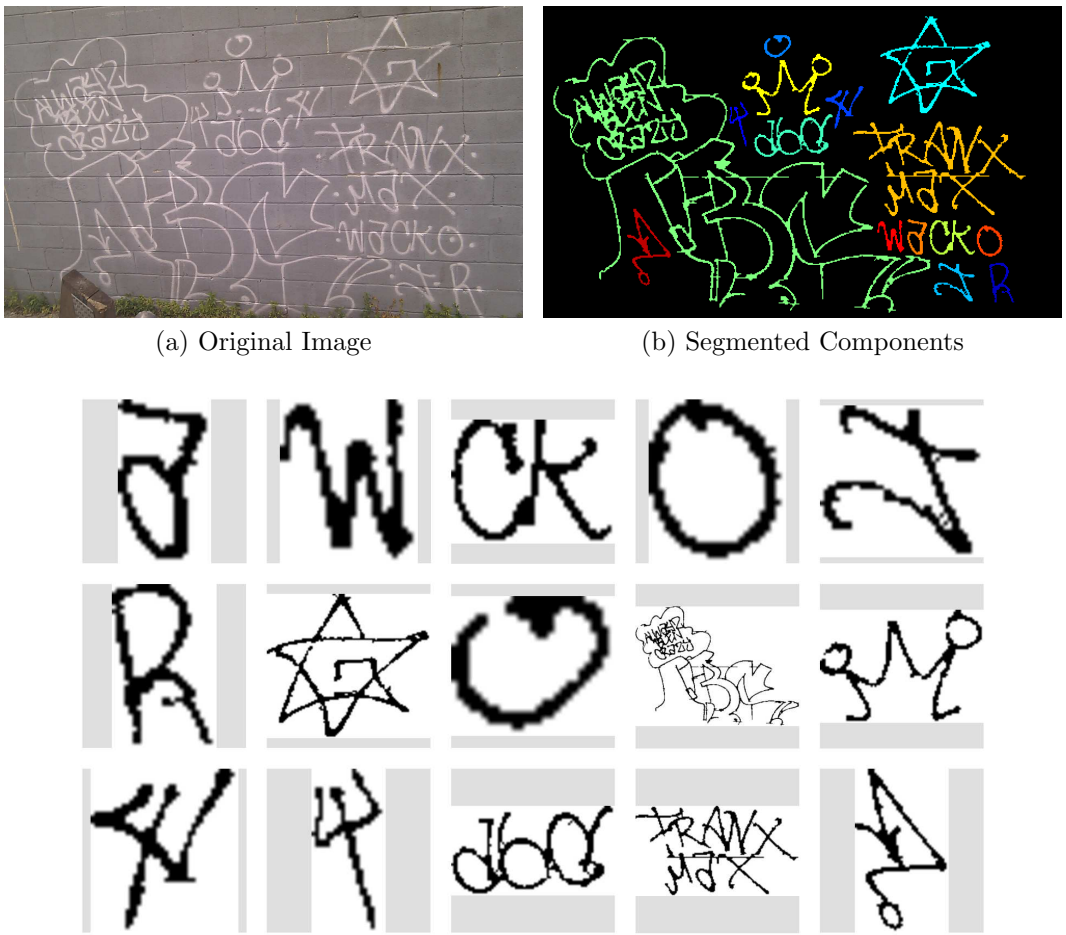
Fig. 5.30.: Number of Connected Components (CCs) Before and After Automatic Gang Graffiti Segmentation.

(a) Segmented Components (b) Merged Components

Fig. 5.31.: Merged Connected Components Forming Words.

Each of the graffiti component candidates are independently classified to return a predicted class and a confidence score. The prediction class corresponds to one of the 14 trained classes, and the confidence score is the score given to the predicted class according to the equations presented in Section 5.1.3, in the range $[0, 1]$. Figures 5.32 to 5.34 show the classification results of one of the test images for each of its components, including component color, predicted class and confidence. Figure 5.35 shows a test image were gang graffiti components are found from two different colors (i.e. two different touchscreen traces). Note how even though one component is sprayed on top of the other we are able to recover the one on the back and successfully classify it. Further automatic interpretation can be done to understand that the component in the back has been crossed-out as a thread from a rival gang. Note that although some graffiti components have been successfully segmented they do not belong to any of the 14 classes we have trained. They are currently assigned to the closest class and given a low confidence score. For symbols that belong to the trained set we usually obtain a confidence higher than 0.60. Therefore, we can discard results if we do not achieve a minimum confidence score.

In the 20 test images there are a total of 98 gang graffiti components; 82 of them can be found in our set of trained classes. We are able to segment and isolate 75 of the 98 gang graffiti components, corresponding to 66 of the 82 recognizable components. We can then successfully classify 59 of them. The segmentation fails when either graffiti components are discarded or multiple graffiti components are merged into one. In all cases we are able to correctly identify the color of the graffiti component based on the median value of the color corrected touchscreen trace. That is, we have an end-to-end gang graffiti accuracy of 71.95%. The accuracy of each of the blocks is as follows: 100% color recognition accuracy, 76.56% automatic segmentation accuracy on color corrected images (80.49% for recognizable components), and 89.39% gang graffiti component classification accuracy on successfully segmented components. Table 5.22 show the accuracies of the automatic segmentation and graffiti component classification steps.

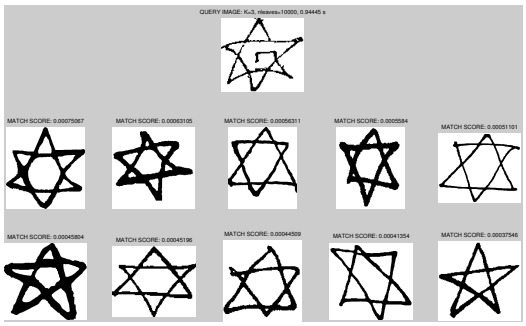


(a) Original Image

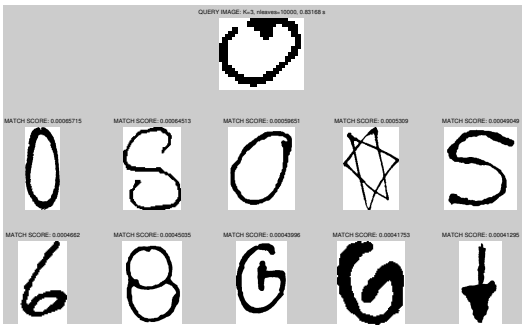
(b) Segmented Components

(c) Graffiti Component Candidates

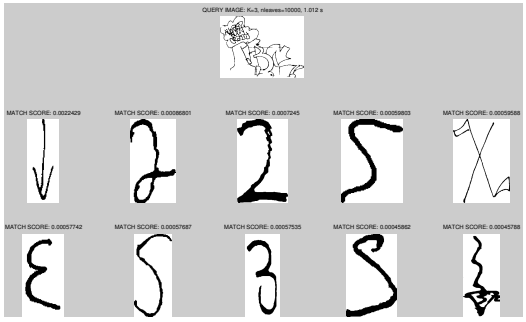
Fig. 5.32.: Automatically Segmented Candidate Graffiti Components.



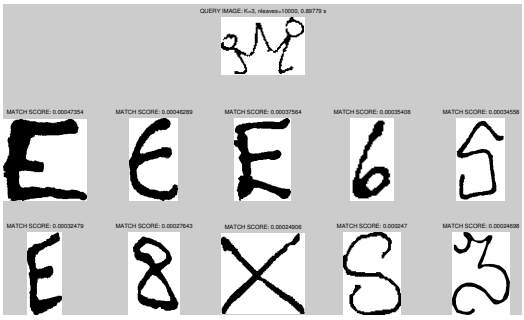
(a) White. *6-Point Star*. Confidence: 0.84



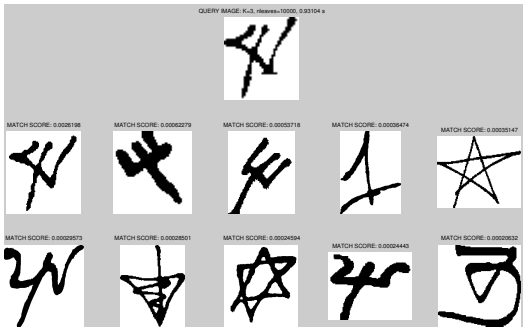
(b) White. *0*. Confidence: 0.26



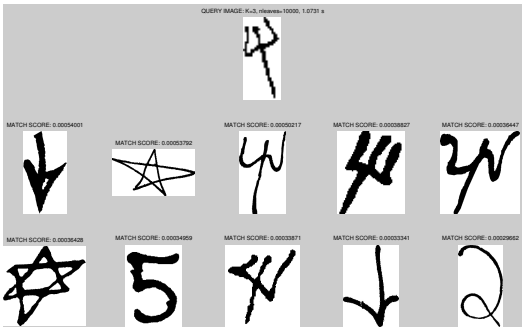
(c) White. *arrow*. Confidence: 0.35



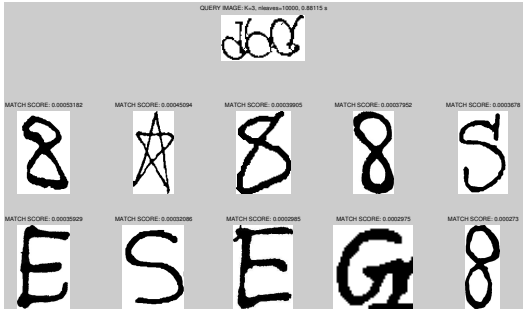
(d) White. *E*. Confidence: 0.49



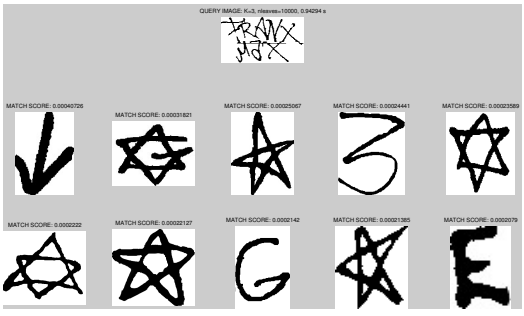
(e) White. *Pitchfork*. Confidence: 0.75



(f) White. *Pitchfork*. Confidence: 0.70

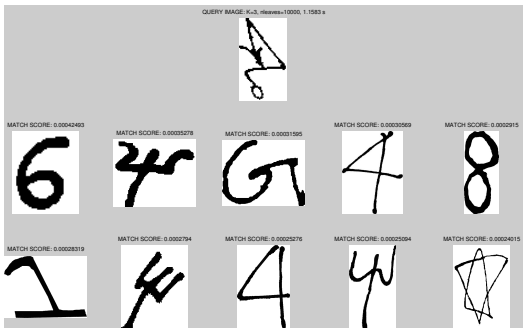


(g) White. *8*. Confidence: 0.43

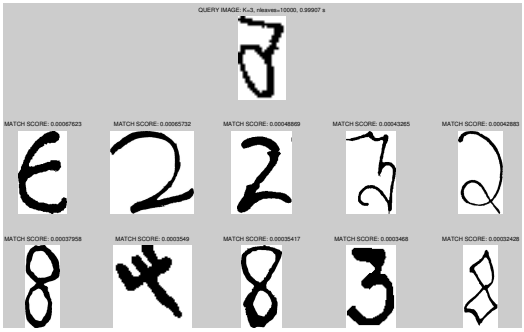


(h) White. *6-Point Star*. Confidence: 0.39

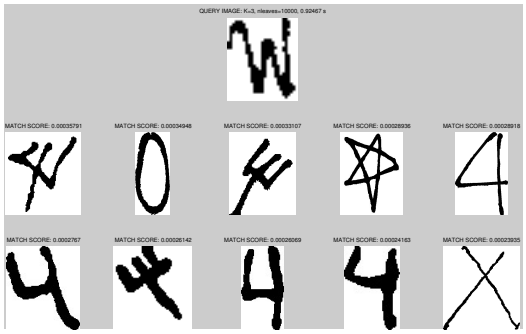
Fig. 5.33.: Classification Results and Top-10 Matches for Candidates 1 to 8.



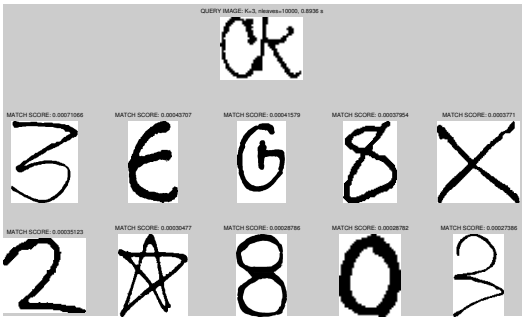
(a) White. *Pitchfork*. Confidence: 0.29



(b) White. *2*. Confidence: 0.35

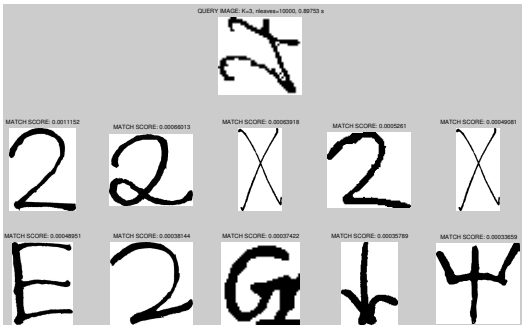


(c) White. *4*. Confidence: 0.37

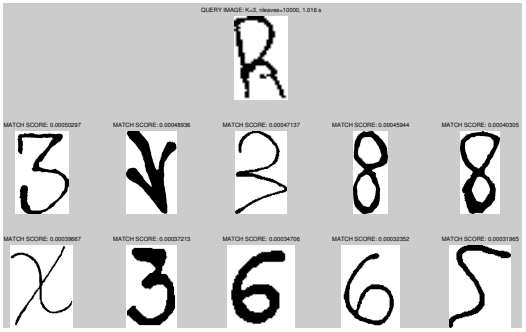


(d) White. *3*. Confidence: 0.26

(e) White. *0*. Confidence: 0.70



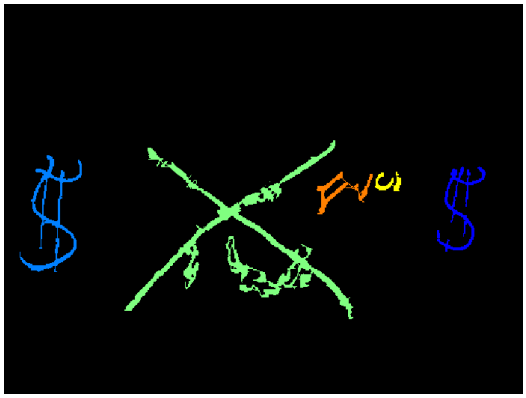
(f) White. *2*. Confidence: 0.50



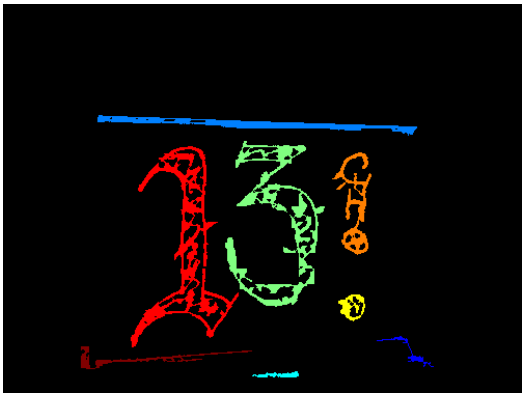
(g) White. *3*. Confidence: 0.33

Fig. 5.34.: Classification Results and Top-10 Matches for Candidates 9 to 15.

(a) Original Image



(b) Segmented Components in Hue



(c) Segmented Components in Luma

(d) Red. *X*. Confidence: 0.72

(e) Black. *1*. Confidence: 0.71

(f) Black. *3*. Confidence: 0.67

Fig. 5.35.: Automatic Segmentation and Classification from Multiple Colors.

Table 5.22: Automatic Segmentation and Graffiti Component Classification Accuracies. N GC: Number of gang graffiti components. N GC Rec: Number of recognizable gang graffiti components.

Image Number	**N GC**	**Segmented**	**N GC Rec**	**Segmented Rec**	**Classified**
1001	6	4	4	2	2
1002	7	7	3	3	3
1003	5	5	4	4	3
1004	2	2	7	6	3
1005	3	3	8	7	6
1006	2	2	1	1	1
1007	2	2	0	0	0
1008	8	4	6	4	3
1009	4	4	3	3	3
1010	8	4	8	7	7
1011	5	5	0	0	0
1012	2	2	0	0	0
1013	7	4	6	3	3
1014	4	3	3	3	2
1015	2	2	2	2	2
1016_1	4	4	3	3	3
1016_2	2	2	2	2	2
1017	5	4	4	4	4
1018	9	1	6	0	0
1019_1	2	2	4	4	4
1019_2	5	5	5	5	5
1020	4	4	3	3	3
Total	**98**	**75**	**82**	**66**	**59**
Accuracy		**76.53%**		**80.49%**	**71.95%**
Marginal Acc					**89.39%**

Table 5.23: Average Running Times (seconds) and Accuracies of The Three Main Blocks in The GARI System on Testing Dataset.

	Color Correction	**Segmentation**	**Classification**	**End-To-End**
Time	1.78	2.04	0.64	4.69
Accuracy	100%	80.49%	89.39%	71.95%

Table 5.23 summarizes the results of the end-to-end system. The Color Correction time is based on the entire image and its accuracy is based on the touchscreen tracing results.

Table 5.24: Number of Images and Users In the Different GARI Systems.

	GARI Classic	GARI IND	GARI CCSO	Total
Images	720	595	173	1,488
Users	73	138	61	272

5.1.5 Database of Gang Graffiti

As of March 2014, our databases of gang graffiti images in the different GARI systems (GARI Classic, GARI IND, GARI CSSO) accumulate 1,488 browsable images with associated thumbnails and reduced size versions, for a total of 1.82 GB of data. We have a total of 272 users from more than 20 departments in the United States distributed across the GARI systems.

We cooperated with the Indianapolis Metropolitan Police Department (IMPD) to acquire a separate set of 657 graffiti images for research purposes. This allows us to be able to accurately calibrate and analyze the images. These include images acquired with and without using a tripod and with and without fiducial markers. We used three digital cameras for this purpose: a 10Mpx Canon Powershot S95, a 4Mpx Panasonic Lumix DMC-FZ4, and a 5Mpx HTC Desire (Android mobile telephone).

Table 5.24 shows the distribution of images and users across the three GARI systems.

5.1.6 Database Query Performance

We tested the elapsed time between sending an image from the hand-held device, using the Android application, and receiving the results of the upload. On the client side, the process includes sending and receiving the image to the server via HTTPS and returning the graffiti image thumbnail and text retrieved to the user. On the server side, the process includes creating a session for the user, checking image existence in the database, copying the image to a specific directory, creating the thumbnail image and reduced size copies of the image, extracting up to 24 EXIF data

points from the image, creating a new entry in the PosgtreSQL table and adding information in as many as 30 fields, and sending back a string with the results of the upload. Table 5.25 shows the details of ten graffiti image uploads using the same network conditions (WiFi). As one can see most of the elapsed time is due to the HTTPS connection since the user interface operations on the hand-held device (for the specific action of uploading an image to the server) do not slow down the process.

Table 5.25: Elapsed Time On the Hand-Held Device and the Server When Uploading an Image.

Image Size	**Server Time**	**Total Time**
146.7 KB	0.66 s	2.24 s
157.9 KB	0.65 s	2.33 s
179.8 KB	0.65 s	2.66 s
203.3 KB	0.66 s	2.42 s
207.9 KB	0.64 s	2.44 s
227.8 KB	0.65 s	2.34 s
609.9 KB	1.05 s	3.64 s
639.8 KB	1.47 s	4.71 s
653.6 KB	1.06 s	4.00 s
760.4 KB	1.07 s	4.31 s

5.2 MERGE[3]

We did experiments for our three proposed methods from Section 4. The first experiment evaluates the accuracy of the sign location detection and color recognition of the segment detection using geometric constraints (see Section 4.2). The second experiment evaluates the accuracy of the sign detection, color recognition, and the saliency map methods of the convex quadrilateral detection based on saliency map (see Section 4.3). The third experiment evaluates the accuracy of the sign location detection of the sign detection based on Fourier descriptors (see Section 4.4). The tests were executed on a desktop computer with a 2.8GHz CPU and 2GB RAM.

[3]The work presented in this section was done by the author jointly with Bin Zhao and Kharittha Thongkor.

The ground-truth information included the sign distance from the camera, sign color, projective distortion of the sign, image resolution, possible shadow affecting the sign, and sign location on the image. Note that we only used the color and not the text of the sign for sign identification for these experiments. The image dataset consisted of 50 images each containing one or more hazmat signs (62 hazmat signs in total). Figure 5.36 illustrates some of the images in the dataset. The images were acquired by first responders using three different cameras: a 8.2 Mpx Kodak Easyshare C813, a 16 Mpx Nikon Coolpix S800c, and a 5 Mpx camera on an HTC Wildfire mobile telephone. The images were acquired in the field, under various lightning conditions, distances, and perspectives. Among the 50 images, 23 were acquired at 10-50 feet, 23 at 50-100 feet, and 4 at 100-200 feet. Among the 62 hazmat signs, 2 had low resolution, 11 had projective distortion, 8 were blurred, and 6 were shaded.

Fig. 5.36.: Example Images From The Test Dataset.

Table 5.26: Analysis Results: Segment Detection Using Geometric Constraints.

Total Signs	**Signs Detected**	**Accuracy**	**Color Recognized**	**Accuracy**
62	22	36.5%	12	19.4%

5.2.1 Segment Detection Using Geometric Constraints

The first experiment consisted of images from a dataset and manually comparing the results with ground-truth information. The method used for this experiment is segment detection using geometric constraints (see Section 4.2). Table 5.26 shows the results of the first experiment using our proposed method. We determined how many signs were successfully detected (*Signs Detected*) and how many were successfully identified (i.e., sign detected plus correct color (*Color Recognized*)). Note that the sign color recognition was done only if a sign was detected. Also note that although this method uses OCR on detected signs, its accuracy was not good enough to be tested on a wide range of images. Among the successfully detected signs we had a higher accuracy for color recognition. The proposed method recognized the correct color in 54.5% of the successfully detected signs. The low accuracy is caused by multiple factors, including segment overlapping, edge detection failure on low resolution images, distortion and rotation of the sign, and multi-colored signs. The proposed method had an average execution time of 2.30 seconds.

5.2.2 Convex Quadrilateral Detection Based on Saliency Map

The second experiment consisted of images from the same dataset from the first experiment, and manually comparing the results with ground-truth information. The method used for this experiment is convex quadrilateral detection based on saliency map (see Section 4.3). We did two experiments to investigate the speed and accuracy of our proposed method. The first experiment consisted of constructing saliency maps using different visual saliency models and evaluating their performance based on ground-truth information. The second experiment consisted of hazmat sign detec-

tion and recognition on our image dataset and manually comparing the results with ground-truth information.

Table 5.27 shows the results of our first experiment, including average execution times and scores. The saliency map methods evaluated in the experiment are: SBVA [312], GBVS [313], DVA [269], MSDA [271], IS [291], HFT [273]. We classified the resulting saliency maps into four categories: good, fair, bad, and lost. For each sign, we assigned 3 points to a good map (sign was mostly contained in a high saliency-valued region), 2 points to a fair map (sign was mostly contained in a middle saliency-valued region), 1 point to a bad map (sign was mostly contained in a low saliency-valued region), and 0 points to a lost map (sign was not contained in any saliency-valued region). Figure 5.37 illustrates examples of each category. The score of each saliency map method is calculated as the sum of the points assigned to all 62 hazmat signs, which ranges from 0 to 186. Compared with the SBVA and the GBVS methods using one color space, the IS and the HFT methods using one color space have comparable scores, while the IS and the HFT methods using two color spaces have higher scores. The IS(RGB+Lab), the HFT(RGB+Lab) and the IS+HFT(RGB+Lab) methods using two color spaces run 2.76, 1.93, and 1.14 times faster than the SBVA method and 4.48, 3.13, and 1.84 times faster than the GBVS method respectively. The results verified that the IS and the HFT methods can be combined to improve the score of IS+HFT method, while still running faster than SBVA and GBVS methods.

Table 5.28 shows the image analysis results of our second experiment. The overall sign detection accuracy is closely related to the number of pixels on a hazmat sign, which is mainly influenced by the distance from a camera in a mobile device to a hazmat sign and the resolution of the image captured by the camera. Compared with the proposed IS(RGB+Lab) and the HFT(RGB+Lab) methods using one saliency map method, our proposed IS+HFT(RGB+Lab) method using two saliency map methods has higher accuracy. The proposed IS+HFT(RGB+Lab) method has an overall sign detection accuracy of 64.5% for all 62 hazmat signs. Note that its

Table 5.27: Average Execution Time (in Seconds), Distribution and Score of Each Saliency Map Method (Color Spaces).

Saliency Map	**Time**	**Good**	**Fair**	**Bad**	**Lost**	**Score**
SBVA(I-RG-BY)	2.07	34	16	11	1	145
GBVS(I-RG-BY)	3.36	30	15	15	2	135
DVA(RGB)	0.43	19	2	11	30	72
MSDA(RGB)	3.74	22	7	27	6	107
IS(I-RG-BY)	0.43	23	4	17	18	94
IS(RGB)	0.36	45	8	4	5	155
IS(Lab)	0.39	27	5	20	10	111
HFT(I-RG-BY)	0.59	33	8	12	9	127
HFT(RGB)	0.53	38	5	8	11	132
HFT(Lab)	0.55	37	10	8	7	139
IS(RGB+Lab)	0.75	52	6	1	3	169
HFT(RGB+Lab)	1.08	41	6	8	7	143
IS+HFT(RGB+Lab)	1.83	55	4	2	1	175

overall accuracy is 71.9% for the 32 hazmat signs in the 50-100 feet range and 50.0% for the 6 hazmat signs in the 100-200 feet range. We can increase the overall accuracy by improving the adaptive thresholding method used in the saliency region segmentation and the morphological operations used in the convex quadrilateral shape detection. We determined the color recognition accuracy based on how many signs were correctly color recognized after a successful sign detection. The color recognition accuracies of the proposed methods using IS(RGB+Lab), HFT(RGB+Lab) and IS+HFT(RGB+Lab) are 37.1%, 30.6%, and 51.6% respectively. Note that the sign color recognition was done only if a sign was successfully detected, and that multi-colored signs may also cause our method to misidentify the sign color, given that we detect signs at individual color channels. Color recognition accuracy is affected by the absence of color calibration in the step of image preprocessing. The overall average execution times of the proposed methods using IS(RGB+Lab), HFT(RGB+Lab) and IS+HFT(RGB+Lab) are 2.60, 2.49, and 5.09 seconds in total respectively. The proposed IS+HFT(RGB+Lab) method is still suitable for real-time applications.

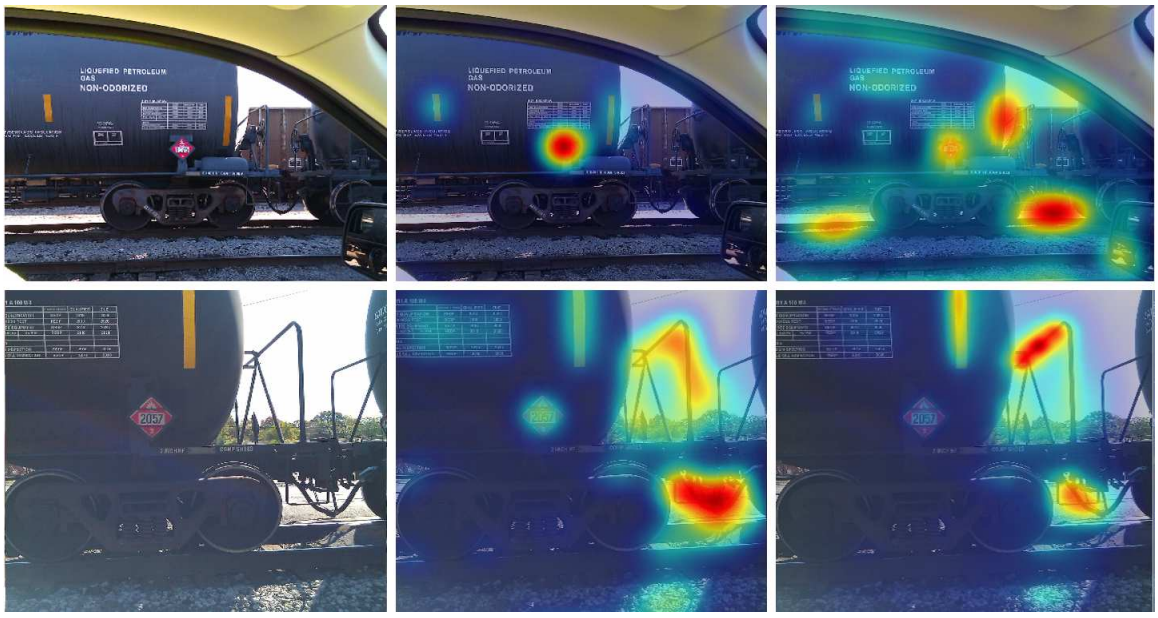


Fig. 5.37.: Saliency map categories (top to bottom, left to right): original image, good, fair; original image, bad, lost.

Table 5.28: Image Analysis Results: Convex Quadrilateral Detection Based on Saliency Map.

Proposed Method	**Total Signs**	**Signs Detected**	**Overall Accuracy**
IS(RGB+Lab)	62	32	51.6%
HFT(RGB+Lab)	62	24	38.7%
IS+HFT(RGB+Lab)	62	40	64.5%

Table 5.29: Analysis Results: Sign Location Detection Based on Fourier Descriptors.

Total Signs	**Signs Detected**	**Accuracy**
62	45	72.6%

Table 5.30: Image Analysis Results for the Three Proposed Methods. 1: Segment Detection Using Geometric Constraints, 2: Convex Quadrilateral Detection Based on Saliency Map, 3: Sign Location Detection Based on Fourier Descriptors.

Proposed Method	**Total Signs**	**Signs Detected**	**Overall Accuracy**	**Time**
1	62	22	36.5%	2.30
2	62	40	64.5%	5.09
3	62	45	72.6%	6.11

5.2.3 Sign Location Detection Based on Fourier Descriptors

We implemented the methods in [257] and our previous technique [314] and compared their accuracy against our method. Table 5.29 shows the results. Our method has a hazmat sign location detection rate of 72.58%, while the detection rates for [257] and [314] are 24.32% and 64.52%, respectively. Figure 5.38 illustrates some examples of sign location detection for each of the methods. The proposed method had an average execution time of 6.11 seconds.

Table 5.30 shows the analysis results for each of the three proposed methods for hazmat sign detection.

Fig. 5.38.: Examples of sign location detection. Column from left to right: results from [257], results from [314], results from proposed method.

6. CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

In this thesis two integrated mobile systems are described. First, a system for gang graffiti image acquisition and recognition. We called this system Gang Graffiti Automatic Recognition and Interpretation or GARI. GARI includes motion blur prevention and detection, color correction based on light sensor, color recognition based on touchscreen tracing, color image segmentation based on Gaussian thresholding, and content-based gang graffiti image retrieval. We have also investigated the design and deployment of an integrated image-based database system. Second, a system for hazmat sign detection and recognition. We called this system Mobile Emergency Response Guidebook or MERGE. MERGE includes segment detection using geometric constraints, convex quadrilateral detection based on saliency map, and sign location detection based on Fourier descriptors.

The main contributions of GARI and MERGE in the area of image analysis are as follows:

- We presented a motion blur prevention and detection method based on mobile device sensors.
- We presented a color correction method based on mobile device light sensor.
- We described a color recognition method based on touchscreen tracing.
- We presented a color image segmentation method based on Gaussian thresholding, block-wise Gaussian segmentation enhancement, background stripe removal, and connected component reconnection.

- We presented a feature extraction method based on local shape context descriptors from SIFT keypoint locations.
- We presented a gang graffiti content based image retrieval method based on bag-of-words model.
- We presented a segment detection method based on geometric constraints.
- We presented a convex quadrilateral detection method based on saliency map.
- We presented a sign location detection based on Fourier descriptors.

The main contributions of GARI and MERGE in the design and deployment of the integrated image-based database system are as follows:

- We developed an integrated image-based database system where data from users and images is connected to gang graffiti information for analysis and tracking.
- We developed an integrated image-based database system where data from users and images is connected to hazmat sign information for image analysis and forensics.
- We created a web-based interface for first responders and researchers to upload images and browse gang related information by location, date and time, using interactive maps for better visualization. It is accessible from any device capable of connecting to the Internet, including iPhone and Blackberry.
- We created a web-based interface for first responders and researchers to upload images and browse hazardous material information by location, date and time for forensic analysis. It is accessible from any device capable of connecting to the Internet, including iPhone and Blackberry.
- We created Android and iOS applications for first responders on the field to upload images to the server, use image analysis and conduct forensic tasks, browse related information, and use location-based services to populate interactive maps.

6.2 Project Status

As of March 2014 we have developed Android and iOS applications and a web-based interface for both the GARI and MERGE systems. The GARI Android/iOS applications include color recognition, image acquisition and upload, content based image retrieval, and database browsing through lists, interactive maps and augmented reality interfaces. The GARI web-based interface includes image upload and database browsing through lists and interactive maps. The MERGE Android/iOS applications include sign recognition and interpretation and internal database browsing using the 2012 version of the Emergency Response Guidebook (ERG). The MERGE web-based interface includes the same capabilities. Both GARI and MERGE web-based interfaces can be accessed from any device capable of connecting to the Internet (e.g., Blackberry, laptop/desktop computers).

Table 6.1 shows the Android/iOS versions of the GARI and MERGE mobile applications as of March 2014. Note that GARI has multiple versions, since it has been deployed for different Police Departments across the country. GARI Classic and GARI Classic Test are versions based at Purdue University and used for testing purposes. GARI IND is used by the Indianapolis Metropolitan Police Department (IMPD). GARI CCSO is used by the Cook County Police Department (CCPD). CGAP stands for Citizen Gang Alert Program. It will be released to the public so regular citizens can report gang graffiti directly to the police.

Table 6.1: Android/iOS versions of the GARI and MERGE mobile applications.

	Android	**iOS**
GARI Classic	2.84 - February 2014	1.3 - November 2013
GARI Classic Test	2.76TEST - February 2014	1.3TEST - November 2013
GARI IND	2.76IND - February 2014	1.4IND - January 2014
GARI CCSO	2.76CCSO - February 2014	1.3CCSO - November 2013
CGAP	1.16 - February 2014	1.3 - November 2013
MERGE	3.0 - February 2014	1.5 - March 2014

Our current image analysis system for GARI includes five methods. First, mobile-based motion blur prevention and detection. Second, color correction based on mobile light sensor. Third, color recognition based on touchscreen tracing. Fourth, automatic graffiti component segmentation, which includes color image segmentation based on Gaussian thresholding, block-wise Gaussian segmentation enhancement, background stripe removal, and graffiti component reconnection. Fifth, content based gang graffiti image retrieval. The first two are done on the client, while the last three are currently done on the server.

Our current image analysis system for MERGE includes three methods, all done on the server. First, segment detection using geometric constraints. Second, convex quadrilateral detection based on saliency map. Third, sign location detection based on Fourier Descriptors.

Our tests on database query performance for GARI suggest that the bottleneck for the upload and retrieval process is from the network connection. This is because we require the full resolution image, which can be up to several MB of data, to be sent to the server for analysis.

Our databases of gang graffiti images in the various GARI systems (GARI Classic, GARI IND, GARI CSSO) have 1,488 browsable images with associated thumbnails and reduced size versions (total of 1.82 GB of data). We have also acquired 657 images for research purposes. The Android and iPhone applications have a memory size of 6.4 MB and 1.7 MB respectively. The CGAP version of the application requires only 1.1 MB and 779 KB respectively.

Our proposed color correction method based on the mobile light sensor has proved to be faster than using fiducial markers and more accurate than using a fiducial marker every week. Our accuracy and speed tests for the content based gang graffiti image retrieval for GARI were done in two scenarios: scene recognition and gang graffiti component classification. The experimental results showed that using SIFT descriptors for scene recognition and LSC descriptors for component classification produce very accurate outcomes. The experiments also showed that the image retrieval is fast

in both scenarios. The end-to-end system has an accuracy of 71.95% and an average execution time of 4.69 seconds as follows: 100% color recognition accuracy, 80.49% automatic segmentation accuracy on color corrected images, and 89.39% gang graffiti component classification accuracy on successfully segmented components.

Our image analysis tests for MERGE showed that the sign location detection based on Fourier Descriptors is more accurate than the convex quadrilateral detection based on saliency map method and the segment detection using geometric constraints. Although it runs slower, its average execution time of 6.11 seconds makes it suitable for real-time operation.

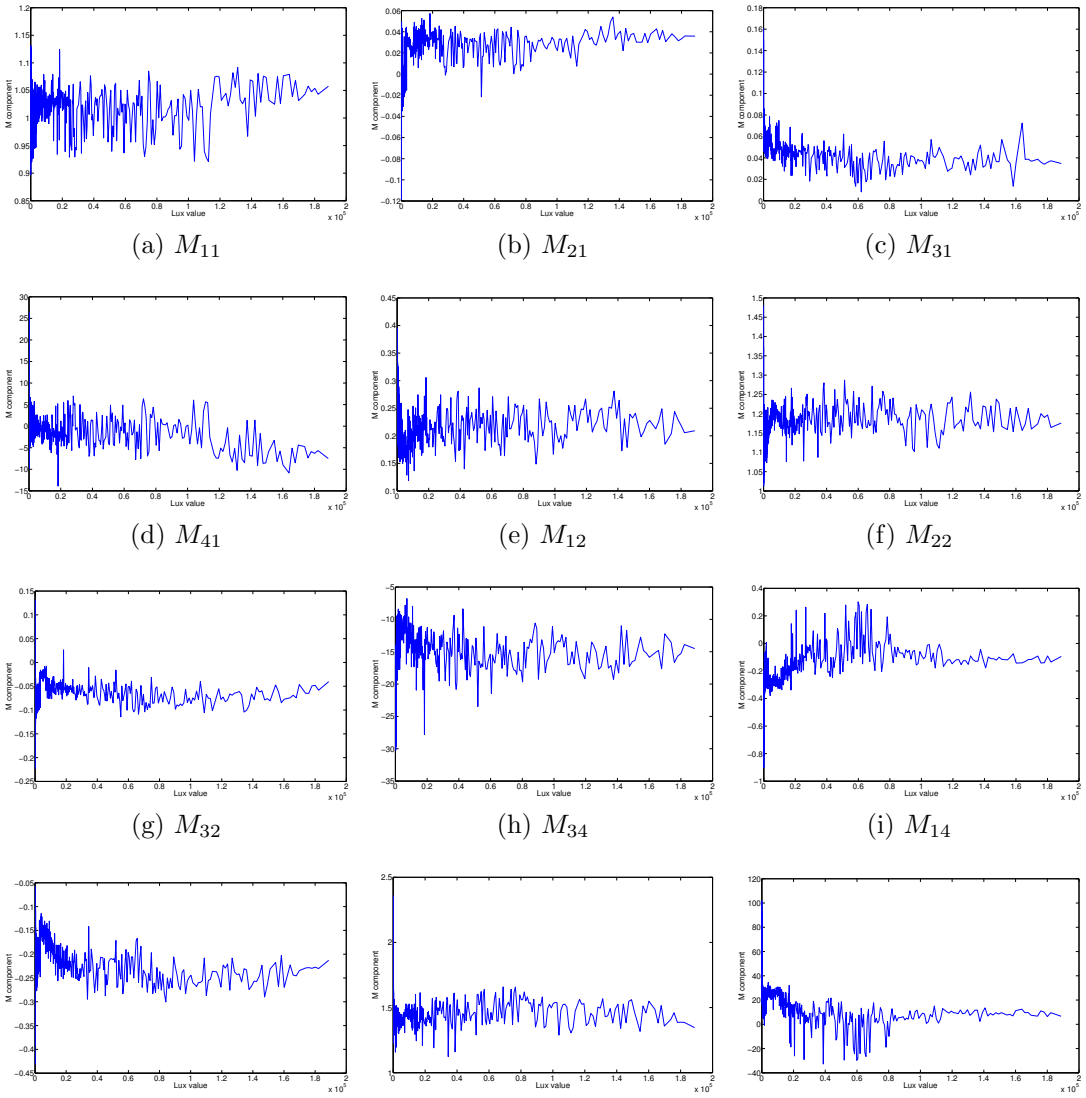
6.3 Future Work

6.3.1 GARI

Although the Color Correction Based on Mobile Light Sensor achieves good accuracy the current method to associate a color correction matrix M to a lux value is through a lookup table. We should investigate automatic generation of color correction matrices from the lux value by describing the evolution of the elements in M with the lightning step. Figure 6.1 illustrates such evolution with the current number of lux samples (612).

Our experiments shown that the bottleneck for the upload and retrieval process is the network connection. Therefore, we could pre-process the image on the mobile device to reduce the amount of data to be sent to the server. In this case we would need to investigate the trade-offs between battery life, network bandwidth, storage capacity, and processor performance [315, 316].

The Block-Wise Gaussian Segmentation Enhancement currently uses a fixed block size for local image processing. In the future we could improve the enhancement by adapting the block size to the local width of the graffiti component. We can use the Stroke Width Transform (SWT) proposed in [311] for this purpose.



(a) M_{11} (b) M_{21} (c) M_{31}

(d) M_{41} (e) M_{12} (f) M_{22}

(g) M_{32} (h) M_{34} (i) M_{14}

(j) M_{24} (k) M_{34} (l) M_{44}

Fig. 6.1.: Evolution of the Elements in M With the Lightning Step (Lux Value).

Our Gang Graffiti Component Classification method is currently able to predict 14 different classes. This is because we want to have at least 15 samples of a particular class to ensure a minimum confidence. In the future, when more images are available from our users we will have more ground-truth samples to extend the number of classes. With more ground-truth data we can also investigate new features for graffiti component classification, such as Zernike moments (global and local) or the curvature scale space descriptors (CSSD) found in MPEG-7.

On the client side, the Android/iPhone users can help improve the classification system by manually correcting the predicted results. The corrections can be sent back to the server and used to automatically retrain the vocabulary tree to account for the changes.

When two or more graffiti components are merged (e.g., 6-point star with pitchforks) we are not currently able to classify them as separate objects. In fact, the new merged component may not be classified as any of the individual sub-components contained in it. We could investigate methods to retrieve multiple objects from a single entity, such as [317] or [318].

Even though our image retrieval methods achieve high accuracy, the procedure to obtain the vocabulary tree involves the segmentation of a high-dimensional space in hierarchical clusters using k-means clustering. This can cause unwanted results due to effects of the curse of dimensionality [237, 238]. We may want to investigate other methods that are more reliable. A tree-like structure can be built by repeatedly projecting the set of $\mathtt{R}^{128}$ descriptors into $\mathtt{R}$ using a normalized random vector $v \in \mathtt{R}^{128}$ until the projection can be clearly separated into two regions or classes. We can use the same method recursively until we obtain the desired number of classes. The resulting tree can then act as a vocabulary tree.

The final output of our current end-to-end system is a list of candidate gang graffiti components and their confidence scores. We can create associations between graffiti components and their descriptions in order to improve the interpretation and help first responders identify gangs, gang members, and track gang activity. However, this is

not as easy as creating a table with one to one correspondences between components and descriptions. Depending on the geographical location of the graffiti the same graffiti component can have different meanings. Although we do not have direct evidence, this may be also true for colors. A more comprehensive database could also include information related to the locations of graffiti components with respect to each other to provide more context information. Also, we can enlarge the number of fields and relationships in the database so as to link gangs to their respective colors, acronyms, gang members, locations, or activity over time.

6.3.2 MERGE

Our long term goal for MERGE is to develop a system based on a mobile device such as a mobile telephone, capable of using location-based services, combined with image analysis, to automatically detect hazardous material signs from images taken up to 500 feet, and provide real-time information to first responders to identify the hazardous materials and determine what specialty equipment, procedures and precautions should be taken in the event of an emergency. This can be done by improving our current sign location detection method and use a more robust color recognition technique. We can also combine the saliency map method from Convex Quadrilateral Detection Based on Saliency Map with the shape descriptors from Sign Location Detection Based on Fourier Descriptors in one method.

We can use the same color correction and blur detection methods from GARI to improve the color recognition and reduce the impact of motion blur. An optical character recognition method would help interpret the text inside the hazmat signs when we have enough image resolution. We can also investigate color recognition methods for multi-colored signs.

6.4 Publications Resulting From This Work

Conference Papers

1. Bin Zhao, **Albert Parra** and Edward J. Delp, "Mobile-Based Hazmat Sign Detection System," *Proceedings of the IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 735-738, December 2013, Austin, TX.

2. **Albert Parra**, Bin Zhao, Joonsoo Kim and Edward J. Delp, "Recognition, Segmentation and Retrieval of Gang Graffiti Images on a Mobile Device," *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, pp. 178-183, November 2013, Waltham, MA.

3. **Albert Parra**, Bin Zhao, Andrew Haddad, Mireille Boutin and Edward J. Delp, "Hazardous Material Sign Detection and Recognition," *Proceedings of the IEEE International Conference on Image Processing*, pp. 2640-2644, September 2013, Melbourne, Australia.

4. **Albert Parra**, Mireille Boutin and Edward J. Delp, "Location-Aware Gang Graffiti Acquisition and Browsing on a Mobile Device," *Proceedings of the IS&T/SPIE Electronic Imaging on Multimedia on Mobile Devices*, pp. 830402-1-13, January 2012, San Francisco, CA.

LIST OF REFERENCES

LIST OF REFERENCES

[1] *ERG*. www.phmsa.dot.gov/hazmat/library/erg

[2] A. Parra, "An integrated mobile system for gang graffiti image acquisition and recognition," M.S. Thesis, Purdue University, West Lafayette, IN, December 2011.

[3] "Graffiti Tracker." graffititracker.net

[4] "Tracking and Automated Graffiti Reporting System." www.594graffiti.com

[5] "Graffiti Reduction & Interception Program." www.gripsystems.org

[6] "Graffiti Tracking System." www.graffititrackingsystem.com

[7] A. K. Jain, J.-E. Lee, and R. Jin, "Graffiti-ID: Matching and retrieval of graffiti images," *Proceedings of the 1st ACM Workshop on Multimedia in Forensics*, pp. 1–6, October 2009, Beijing, China.

[8] W. Tong, J.-E. Lee, R. Jin, and A. K. Jain, "Gang and moniker identification by graffiti matching," *Proceedings of the 3rd ACM Workshop on Multimedia in Forensics and Intelligence*, pp. 1–6, November 2011, Scottsdale, AZ.

[9] A. Jain, J. Lee, and R. Jin, "Tattoo-ID: Automatic tattoo image retrieval for suspect and victim identification," *Advances in Multimedia Information Processing, PCM*, pp. 256–265, December 2007, Hong Kong, China.

[10] J.-E. Lee, A. Jain, and R. Jin, "Scars, marks and tattoos (SMT): Soft biometric for suspect and victim identification," *Proceedings of the Biometrics Symposium*, pp. 1–8, September 2008, Tampa, FL.

[11] A. K. Jain, J.-E. Lee, R. Jin, and N. Gregg, "Content-based image retrieval: An application to tattoo images," *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pp. 2745–2748, November 2009, Cairo, Egypt.

[12] J.-E. Lee, R. Jin, A. K. Jain, and W. Tong, "Image retrieval in forensics: Tattoo image database application," *IEEE Transactions on Multimedia*, vol. 19, no. 1, pp. 40–49, 2012.

[13] A. Jain, R. Jin, and J.-E. Lee, "Tattoo image matching and retrieval," *IEEE Transactions on Computers*, vol. 45, no. 5, pp. 93–96, May 2012.

[14] H. Han and A. Jain, "Tattoo based identification: Sketch to image matching," *Proceedings of the International Conference on Biometrics (ICB)*, pp. 1–8, June 2013, Madrid, Spain.

[15] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, November 2004, Hingham, MA.

[16] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1090–1104, October 2000, Los Alamitos, CA.

[17] C. Yang, P. C. Wong, W. Ribarsky, and J. Fan, "Efficient graffiti image retrieval," *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, pp. 36:1–36:8, June 2012, Hong Kong, China.

[18] D. Manger, "Large-scale tattoo image retrieval," *Proceedings of the Conference on Computer and Robot Vision*, pp. 454–459, May 2012, Toronto, Canada.

[19] M. Zarem, E. Vuillermet, and J. DeAguiar, "Intelligent reverse geocoding," August 2007, US Patent App. 11/367,911.

[20] W. Niblack, *An Introduction to Digital Image Processing*. Prentice-Hall, 1986.

[21] WISER. `wiser.nlm.nih.gov`

[22] D. Gossow, J. Pellenz, and D. Paulus, "Danger sign detection using color histograms and SURF matching," *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics*, pp. 13–18, October 2008, Sendai, Japan.

[23] J. Meyer, P. Schnitzspan, S. Kohlbrecher, K. Petersen, M. Andriluka, O. Schwahn, U. Klingauf, S. Roth, B. Schiele, and O. Stryk, "A semantic world model for urban search and rescue based on heterogeneous sensors," *Proceedings of the 14th RoboCup International Symposium*, vol. 6556, pp. 180–193, June 2010, Singapore, Singapore.

[24] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-up robust features (SURF)," *Journal of Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, June 2008.

[25] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 886–893, June 2005, San Diego, CA.

[26] GARI. `www.gang-graffiti.org`

[27] MERGE. `www.hazmat-signs.org`

[28] National Gang Intelligence Center (NGIC), *2011 National Gang Threat Assessment - Emerging Trends*. United States Deptartment of Justice, April 2011.

[29] National Drug Intelligence Center (NDIC), *Attorney General's Report to Congress on the Growth of Violent Street Gangs in Suburban Areas*. United States Deptartment of Justice, April 2008.

[30] J. Kim, A. Parra, and E. J. Delp, "Tattoo image matching using local and global shape context," *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, October 2014, Paris, France (submitted).

[31] Japan Electronic Industry Development Association (JEIDA), "Design rule for camera file system, version 1.0." 1998.

[32] D. Ley and R. Cybriwsky, "Urban graffiti as territorial markers," *Annals of the Association of American Geographers*, vol. 64, no. 4, pp. 491–505, December 1974.

[33] J. Ferrell, *Crimes of Style: Urban Graffiti and the Politics of Criminality*. Garland, New York, 1993.

[34] W. Miller, *Crime by Youth Gangs and Groups in the United States*. U.S. Dept. of Justice, Office of Justice Programs, Office of Juvenile Justice and Delinquency Prevention, 1992.

[35] United States Department of Transportation, *Code of Federal Regulations, Title 49, DOT Hazmat*. Labelmaster, October 2012.

[36] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Processing Letters*, vol. 9, no. 3, pp. 81–84, 2002.

[37] R. L. Lagendijk and J. Biemond, *The Image and Video Processing Handbook*. Academic Press, 1999, ch. Basic methods for image restoration and identification, pp. 125–139.

[38] R. Y. Landge and R. Sharma, "Blur detection methods for digital images - A survey," *International Journal of Computer Applications Technology and Research*, vol. 2, no. 4, pp. 494–498, 2013.

[39] J. Ko and C. Kim, "Low cost blur image detection and estimation for mobile devices," *Proceedings of the International Conference on Advanced Communication Technology*, vol. 03, pp. 1605–1610, February 2009, Phoenix Park, Ireland.

[40] B. Cardani, "Optical image stabilization for digital cameras," *IEEE Transacions on Control Systems*, vol. 26, no. 2, pp. 21–22, April 2006.

[41] J.-H. Moon and S. Y. Jung, "Implementation of an image stabilization system for a small digital camera," *IEEE Transactions on Consumer Electronics*, vol. 54, no. 2, pp. 206–212, May 2008.

[42] S. Nasiri, M. Kiadeh, Y. Zheng, S. Lin, and S. Shi, "Optical image stabilization in a digital still camera or handset," May 2012, US Patent 8,170,408.

[43] A. Ciancio, A. L. N. T. da Costa, E. A. B. Da Silva, A. Said, R. Samadani, and P. Obrador, "No-reference blur assessment of digital pictures based on multifeature classifiers," *IEEE Transactions on Image Processing*, vol. 20, no. 1, pp. 64–75, January 2011.

[44] C. Xu, N. Khanna, C. J. Boushey, and E. J. Delp, "Low complexity image quality measures for dietary assessment using mobile devices," *Proceedings of the IEEE International Symposium on Multimedia (ISM)*, pp. 351–356, December 2011, Dana Point, CA.

[45] X. Marichal, W. Ma, and H. Zhang, "Blur determination in the compressed domain using DCT information," *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, vol. 2, pp. 386–390, October 1999, Kobe, Japan.

[46] N. Ahmed, T. Natarajan, and K. Rao, "Discrete Cosine Transform," *IEEE Transactions on Computers*, vol. C-23, no. 1, pp. 90–93, January 1974.

[47] H. Tong, M. Li, H. Zhang, and C. Zhang, "Blur detection for digital images using wavelet transform," *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, vol. 1, pp. 17–20, June 2004, Taipei, Taiwan.

[48] P. Porwik and A. Lisowska, "The haar-wavelet transform in digital image processing: its status and achievements," *Machine graphics and vision*, vol. 13, no. 1/2, pp. 79–98, 2004.

[49] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi, "Perceptual blur and ringing metrics: Application to JPEG2000," *Signal Processing and Image Communication*, vol. 19, no. 2, pp. 163–172, 2004.

[50] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. Wiley-Interscience, 1973.

[51] N. Narvekar and L. Karam, "A no-reference image blur metric based on the cumulative probability of blur detection (CPBD)," *IEEE Transactions on Image Processing*, vol. 20, no. 9, pp. 2678–2683, March 2011.

[52] O. Šindelář and F. Šroubek, "Image deblurring in smartphone devices using built-in inertial measurement sensors," *Journal of Electronic Imaging*, vol. 22, no. 1, pp. 011 003:1–011 003:8, 2013.

[53] P. R. Sanketi and J. M. Coughlan, "Anti-blur feedback for visually impaired users of smartphone cameras," *Proceedings of the 12th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 233–234, 2010, Orlando, FL.

[54] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Transactions on Computer Graphics and Applications*, vol. 21, no. 5, pp. 34–41, September 2001.

[55] G. Sharma and R. Bala, *Digital color imaging handbook*. CRC press, 2002.

[56] A. Gijsenij, T. Gevers, and J. van de Weijer, "Computational color constancy: Survey and experiments," *IEEE Transactions on Image Processing*, vol. 20, no. 9, pp. 2475–2489, September 2011.

[57] M. Bleier, C. Riess, S. Beigpour, E. Eibenberger, E. Angelopoulou, T. Troger, and A. Kaup, "Color constancy and non-uniform illumination: Can existing algorithms work?" *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 774–781, November 2011, Barcelona, Spain.

[58] K. Barnard, L. Martin, A. Coath, and B. Funt, "A comparison of computational color constancy algorithms - Part II: Experiments with image data," *IEEE Transactions on Image Processing*, vol. 11, no. 9, pp. 985–996, September 2002.

[59] G. Buchsbaum, "A spatial processor model for object colour perception," *Journal of the Franklin Institute*, vol. 310, no. 1, pp. 1–26, 1980.

[60] J. van de Weijer, T. Gevers, and A. Gijsenij, "Edge-based color constancy," *IEEE Transactions on Image Processing*, vol. 16, no. 9, pp. 2207–2214, September 2007.

[61] C. Xu, F. Zhu, N. Khanna, C. J. Boushey, and E. J. Delp, "Image enhancement and quality measures for dietary assessment using mobile devices," *Proceedings of the IS&T/SPIE Conference on Computational Imaging X*, vol. 8296, pp. 82 960Q–1–82 960Q–10, January 2012, San Francisco, CA.

[62] S. Srivastava, C. Xu, and E. J. Delp, "White synthesis with user input for color balancing on mobile camera systems," *Proceedings of the IS&T/SPIE Conference on Multimedia on Mobile Devices*, vol. 8304, pp. 83 040F:1–83 040F:8, January 2012, Burlingame, CA.

[63] D. A. Forsyth, "A novel algorithm for color constancy," *International Journal of Compututer Vision*, vol. 5, no. 1, pp. 5–36, August 1990.

[64] G. Finlayson, "Color in perspective," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, pp. 1034–1038, October 1996.

[65] G. Finlayson and S. Hordley, "Improving gamut mapping color constancy," *IEEE Transactions on Image Processing*, vol. 9, no. 10, pp. 1774–1783, October 2000.

[66] K. Barnard, V. Cardei, and B. Funt, "A comparison of computational color constancy algorithms - Part I: Methodology and experiments with synthesized data," *IEEE Transactions on Image Processing*, vol. 11, no. 9, pp. 972–984, September 2002.

[67] H. Joze and M. Drew, "White patch gamut mapping colour constancy," *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pp. 801–804, September 2012, Orlando, FL.

[68] G. Finlayson, S. Hordley, and P. Hubel, "Color by correlation: a simple, unifying framework for color constancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1209–1221, November 2001.

[69] C. Rosenberg, M. Hebert, and S. Thrun, "Color constancy using KL-divergence," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, vol. 1, pp. 239–246, July 2001, Vancouver, Canada.

[70] G. Sapiro, "Color and illuminant voting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 11, pp. 1210–1215, November 1999.

[71] P. Gehler, C. Rother, A. Blake, T. Minka, and T. Sharp, "Bayesian color constancy revisited," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, June 2008, Anchorage, AK.

[72] S. Beigpour, C. Riess, J. van de Weijer, and E. Angelopoulou, "Multi-illuminant estimation with conditional random fields," *IEEE Transactions on Image Processing*, vol. 23, no. 1, pp. 83–96, January 2014.

[73] M. Sajjaa and G. Fischer, "Automatic white balance: WhitebalPR using the dichromatic reflection model," *Proceedings of the IS&T/SPIE Conference on Digital Photography*, vol. 7250, pp. 72 500D–72 500D–12, January 2009, San Jose, CA.

[74] F. Zaraga and G. Langfelder, "White balance by tunable spectral responsivities," *Journal of the Optical Society of America*, vol. 27, no. 1, pp. 31–39, January 2010.

[75] A. Ilie and G. Welch, "Ensuring color consistency across multiple cameras," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, vol. 2, pp. 1268–1275, October 2005, Beijing, China.

[76] E. A. Johnson, "Touch display - A novel input/output device for computers," *Electronics Letters*, vol. 1, no. 8, p. 219, 1965.

[77] N. Matsushita and J. Rekimoto, "HoloWall: Designing a finger, hand, body, and object sensitive wall," *Proceedings of the 10th Annual ACM Symposium on User interface Software and Technology*, pp. 209–210, October 1997, Banff, Canada.

[78] S. Izadi, H. Brignull, T. Rodden, Y. Rogers, and M. Underwood, "Dynamo: A public interactive surface supporting the cooperative sharing and exchange of media," *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, pp. 159–168, November 2003, Vancouver, Canada.

[79] J. Rekimoto, "SmartSkin: An infrastructure for freehand manipulation on interactive surfaces," *Proceedings of the 20th Annual SIGCHI Conference on Human Factors in Computing Systems*, pp. 113–120, April 2002, Minneapolis, MN.

[80] A. Pirhonen, S. Brewster, and C. Holguin, "Gestural and audio metaphors as a means of control for mobile devices," *Proceedings of the 20th Annual SIGCHI Conference on Human Factors in Computing Systems*, pp. 291–298, April 2002, Minneapolis, MN.

[81] E. Hoggan, S. A. Brewster, and J. Johnston, "Investigating the effectiveness of tactile feedback for mobile touchscreens," *Proceedings of the 26th Annual SIGCHI Conference on Human Factors in Computing Systems*, pp. 1573–1582, April 2008, Florence, Italy.

[82] K. S. Deoras, M. R. Wolfson, R. L. Searls, S. R. Hilfer, J. B. Sheffield, and T. H. Shaffer, "Use of a touch sensitive screen and computer assisted image analysis for quantitation of developmental changes in pulmonary structure," *Pediatr Pulmonol*, vol. 9, no. 2, pp. 109–18, 1990.

[83] J. Dai and C.-K. Chung, "Touchscreen everywhere: On transferring a normal planar surface to a touch-sensitive display," *IEEE Transactions on Cybernetics*, vol. PP, no. 99, pp. 1–14, November 2013.

[84] J. Krauskopf and G. Karl, "Color discrimination and adaptation," *Vision Research*, vol. 32, no. 11, pp. 2165–2175, January 1992.

[85] K.-M. Cho, J.-H. Jang, and K.-S. Hong, "Adaptive skin-color filter," *Pattern Recognition*, vol. 34, no. 5, pp. 1067–1073, May 2001.

[86] R. Jusoh, N. Hamzah, M. Marhaban, and N. Alias, "Skin detection based on thresholding in RGB and hue component," *Proceedings of the 2010 IEEE Symposium on Industrial Electronics Applications*, pp. 515–517, October 2010, Penang, Malaysia.

[87] R. M. Haralick and L. G. Shapiro, "Image segmentation techniques," *Computer Vision, Graphics, and Image Processing*, vol. 29, no. 1, pp. 100–132, 1985.

[88] K. Fu and J. Mui, "A survey on image segmentation," *Pattern Recognition*, vol. 13, no. 1, pp. 3–16, 1981.

[89] A. Rosenfeld and A. Kak, *Digital Picture Processing Vol. 2.* Academic Press, New York, 1982.

[90] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recognition*, vol. 26, no. 9, pp. 1277–1294, 1993.

[91] T. Q. Chen, Y. L. Murphey, R. Karlsen, and G. Gerhart, "Color image segmentation in color and spatial domain," *Proceedings of the 16th International Conference on Developments in Applied Artificial Intelligence*, pp. 72–82, June 2003, Laughborough, United Kingdom.

[92] W. Skarbek and A. Koschan, "Colour image segmentation - A survey," Technical University of Berlin, Department of Computer Science, Tech. Rep., 1994.

[93] H. Cheng, X. Jiang, Y. Sun, and J. Wang, "Color image segmentation: Aadvances and prospects," *Pattern Recognition*, vol. 34, no. 12, pp. 2259–2281, 2001.

[94] L. Lucchese and S. Mitra, "Color image segmentation: A state-of-the-art survey," *Proceedings of the Indian National Science Academy*, vol. 67 A, pp. 207–221, March 2001, New Delhi, India.

[95] G. Dong and M. Xie, "Color clustering and learning for image segmentation based on neural networks," *IEEE Transactions on Neural Networks*, vol. 16, no. 4, pp. 925–936, July 2005.

[96] Y. He, N. Khanna, C. Boushey, and E. Delp, "Image segmentation for image-based dietary assessment: A comparative study," *Proceedings of the International Symposium on Signals, Circuits and Systems (ISSCS)*, pp. 1–4, July 2013, Iasi, Romania.

[97] S. R. Vantaram and E. Saber, "Survey of contemporary trends in color image segmentation," *Journal of Electronic Imaging*, vol. 21, no. 4, pp. 040 901–1–040 901–28, October 2012.

[98] R. Tan and K. Ikeuchi, "Separating reflection components of textured surfaces using a single image," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 2, pp. 178–193, February 2005.

[99] G. Healey, "Segmenting images using normalized color," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, pp. 64–73, January 1992.

[100] B. A. Maxwell and S. A. Shafer, "Physics-based segmentation of complex objects using multiple hypotheses of image formation," *Computer Vision and Image Understanding*, vol. 65, no. 2, pp. 269–295, November 1997.

[101] F. Jurie and B. Triggs, "Creating efficient codebooks for visual recognition," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 1, pp. 604–610, October 2005, Montbonnot, France.

[102] Y. Tarabalka, J. Benediktsson, and J. Chanussot, "Spectral-spatial classification of hyperspectral imagery based on partitional clustering techniques," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 8, pp. 2973–2987, August 2009.

[103] K.-C. Lee, J. Ho, M.-H. Yang, and D. Kriegman, "Video-based face recognition using probabilistic appearance manifolds," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 313–320, June 2003, Urbana, IL.

[104] A. Fred and A. Jain, "Combining multiple clusterings using evidence accumulation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 6, pp. 835–850, June 2005.

[105] H. Gomez-Moreno, S. Maldonado-Bascon, P. Gil-Jimenez, and S. Lafuente-Arroyo, "Goal evaluation of segmentation algorithms for traffic sign recognition," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 4, pp. 917–930, December 2010.

[106] S. Phung, A. Bouzerdoum, and S. Chai, D., "Skin segmentation using color pixel classification: analysis and comparison," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 1, pp. 148–154, January 2005.

[107] C.-I. Chang, Y. Du, J. Wang, S.-M. Guo, and P. Thouin, "Survey and comparative analysis of entropy and relative entropy thresholding techniques," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 153, no. 6, pp. 837–850, December 2006.

[108] J. Sun, N.-N. Zheng, and H.-Y. Shum, "Stereo matching using belief propagation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 787–800, July 2003.

[109] V. Grau, A. U. J. Mewes, M. Alcaniz, R. Kikinis, and S. Warfield, "Improved watershed transform for medical image segmentation using prior information," *IEEE Transactions on Medical Imaging*, vol. 23, no. 4, pp. 447–458, April 2004.

[110] A. Round, A. Duller, and P. Fish, "Colour segmentation for lesion classification," *Proceedings of the 19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 2, pp. 582–585, November 1997, Chicago, IL.

[111] Y. Deng and B. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 8, pp. 800–810, August 2001.

[112] M. Plissiti, D. Fotiadis, L. Michalis, and G. Bozios, "An automated method for lumen and media-adventitia border detection in a sequence of ivus frames," *IEEE Transactions on Information Technology in Biomedicine*, vol. 8, no. 2, pp. 131–141, June 2004.

[113] N. Funakubo, "Feature extraction of color texture using neural networks for region segmentation," *Proceedings of the 20th Annual Conference of IEEE Industrial Electronics*, vol. 2, pp. 852–856, September 1994, Bologna, Italy.

[114] T. Carron and P. Lambert, "Color edge detector using jointly hue, saturation and intensity," *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, vol. 3, pp. 977–981, November 1994, Austin, TX.

[115] T. Chan and L. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, February 2001.

[116] Y. He, N. Khanna, C. J. Boushey, and E. Delp, "Snakes assisted food image segmentation," *Proceedings of the IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pp. 181–185, September 2012, Banff, Canada.

[117] Y. He, C. Xu, N. Khanna, C. Boushey, and E. Delp, "Food image analysis: Segmentation, identification and weight estimation," *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, July 2013, San Jose, CA.

[118] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal Of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988.

[119] I. Milevskiy and J.-Y. Ha, "A fast algorithm for korean text extraction and segmentation from subway signboard images utilizing smartphone sensors." *Journal of Computing Science and Engineering*, vol. 5, no. 3, pp. 161–166, September 2011.

[120] D. H. Rao and P. Panduranga, "A survey on image enhancement techniques: Classical spatial filter, neural network, cellular neural network, and fuzzy filter," *Proceedings of the IEEE International Conference on Industrial Technology (ICIT)*, pp. 2821–2826, December 2006, Mumbai, India.

[121] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: a technical overview," *IEEE Signal Processing Magazine*, vol. 20, no. 3, pp. 21–36, May 2003.

[122] M. Alam, J. Bognar, R. Hardie, and B. Yasuda, "Infrared image registration and high-resolution reconstruction using multiple translationally shifted aliased video frames," *IEEE Transactions on Instrumentation and Measurement*, vol. 49, no. 5, pp. 915–923, October 2000.

[123] M. Chabert and B. Lacaze, "Non uniform sampling for remote sensing images," *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 4718–4721, July 2012, Munich, Germany.

[124] A. Patti and Y. Altunbasak, "Artifact reduction for set theoretic super resolution image reconstruction with edge adaptive constraints and higher-order interpolants," *IEEE Transactions on Image Processing*, vol. 10, no. 1, pp. 179–186, January 2001.

[125] J. J. Zou and H. Yan, "A deblocking method for BDCT compressed images based on adaptive projections," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 3, pp. 430–435, March 2005.

[126] M. Elad and A. Feuer, "Superresolution restoration of an image sequence: adaptive filtering approach," *IEEE Transactions on Image Processing*, vol. 8, no. 3, pp. 387–395, March 1999.

[127] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, February 2007.

[128] H. Kong, J.-Y. Audibert, and J. Ponce, "General road detection from a single image," *IEEE Transactions on Image Processing*, vol. 19, no. 8, pp. 2211–2220, August 2010.

[129] D. Rajan and S. Chaudhuri, "Simultaneous estimation of super-resolved scene and depth map from low resolution defocused observations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1102–1117, September 2003.

[130] H. Aly and E. Dubois, "Image up-sampling using total-variation regularization with a new observation model," *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp. 1647–1659, October 2005.

[131] F. Salem and A. Yagle, "Non-parametric super-resolution using a bi-sensor camera," *IEEE Transactions on Multimedia*, vol. 15, no. 1, pp. 27–40, January 2013.

[132] N. Nguyen, P. Milanfar, and G. Golub, "Efficient generalized cross-validation with applications to parametric image restoration and resolution enhancement," *IEEE Transactions on Image Processing*, vol. 10, no. 9, pp. 1299–1308, September 2001.

[133] F. Sroubek, G. Cristobal, and J. Flusser, "A unified approach to superresolution and multichannel blind deconvolution," *IEEE Transactions on Image Processing*, vol. 16, no. 9, pp. 2322–2332, September 2007.

[134] E. Faramarzi, D. Rajan, and M. Christensen, "Unified blind method for multi-image super-resolution and single/multi-image blur deconvolution," *IEEE Transactions on Image Processing*, vol. 22, no. 6, pp. 2101–2114, June 2013.

[135] C.-T. Lin, K.-W. Fan, H.-C. Pu, S.-M. Lu, and S.-F. Liang, "An HVS-directed neural-network-based image resolution enhancement scheme for image resizing," *IEEE Transactions on Fuzzy Systems*, vol. 15, no. 4, pp. 605–615, August 2007.

[136] D. Marin, A. Aquino, M. Gegundez-Arias, and J. Bravo, "A new supervised method for blood vessel segmentation in retinal images by using gray-level and moment invariants-based features," *IEEE Transactions on Medical Imaging*, vol. 30, no. 1, pp. 146–158, January 2011.

[137] D. Van De Ville, M. Nachtegael, D. Van der Weken, E. Kerre, W. Philips, and I. Lemahieu, "Noise reduction by fuzzy image filtering," *IEEE Transactions on Fuzzy Systems*, vol. 11, no. 4, pp. 429–436, August 2003.

[138] S. Schulte, M. Nachtegael, V. De Witte, D. Van der Weken, and E. Kerre, "A fuzzy impulse noise detection and reduction method," *IEEE Transactions on Image Processing*, vol. 15, no. 5, pp. 1153–1162, May 2006.

[139] M. Selvi and A. George, "FBFET: Fuzzy based fingerprint enhancement technique based on adaptive thresholding," *Proceedings of the International Conference on Computing, Communications and Networking Technologies (ICCCNT)*, pp. 1–5, July 2013, Tiruchengode, India.

[140] T. Shih, L. Lin, and W. Lee, "Detection and removal of long scratch lines in aged films," *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pp. 477–480, July 2006, Toronto, Canada.

[141] Y.-T. Kao, T. Shih, H.-Y. Zhong, and L.-K. Dai, "Scratch line removal on aged films," *Proceedings of the 9th IEEE International Symposium on Multimedia*, pp. 147–151, December 2007, Taichung, Taiwan.

[142] Z. Qingyue and D. Youdong, "Scratch line detection and restoration based on canny operator," *Proceedings of the Asia-Pacific Conference on Information Processing (APCIP)*, vol. 2, pp. 148–151, July 2009, Shenzhen, Hong Kong.

[143] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 417–424, 2000, New Orleans, LA.

[144] Q. Miao, P. Xu, T. Liu, Y. Yang, J. Zhang, and W. Li, "Linear feature separation from topographic maps using energy density and the shear transform," *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1548–1558, April 2013.

[145] N. I. N. Ismail and A. M. S. Noor, *A Novel Technique for Contour Reconstruction to DEM*, ser. Research Monograph. Pusat Pengurusan Penyelidikan, Universiti Teknologi Malaysia, 2009.

[146] E. Hancer and R. Samet, "Advanced contour reconnection in scanned topographic maps," *Proceedings of the International Conference on Application of Information and Communication Technologies (AICT)*, pp. 1–5, October 2011, Baku, Azerbaijan.

[147] R. Samet and E. Hancer, "A new approach to the reconstruction of contour lines extracted from topographic maps," *Journal of Visual Communication and Image Representation*, vol. 23, no. 4, pp. 642–647, May 2012.

[148] A. N. G. L. Filho and C. A. B. Mello, "A novel method for reconstructing degraded digits," *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 733–738, October 2012, Seoul, South Korea.

[149] A. N. G. L. Filho and C. A. B. Mello, "Degraded digit restoration based on physical forces," *Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR)*, pp. 195–199, August 2013, Washington, DC.

[150] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349–1380, December 2000.

[151] M.-H. Yang, D. Kriegman, and N. Ahuja, "Detecting faces in images: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34–58, Jan 2002.

[152] A. Natsev, R. Rastogi, and K. Shim, "WALRUS: a similarity retrieval algorithm for image databases," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 3, pp. 301–316, March 2004.

[153] P. Hiremath and J. Pujari, "Content based image retrieval using color, texture and shape features," *Proceedings of the International Conference on Advanced Computing and Communications*, pp. 780–784, December 2007, Guwahati, India.

[154] J. Wang and Y. Yagi, "Integrating color and shape-texture features for adaptive real-time object tracking," *IEEE Transactions on Image Processing*, vol. 17, no. 2, pp. 235–240, February 2008.

[155] T. Deselaers, D. Keysers, and H. Ney, "Features for image retrieval: An experimental comparison," *Journal of Information Retrieval*, vol. 11, no. 2, pp. 77–107, April 2008.

[156] Y. Cai and G. Baciu, "Detecting, grouping, and structure inference for invariant repetitive patterns in images," *IEEE Transactions on Image Processing*, vol. 22, no. 6, pp. 2343–2355, June 2013.

[157] O. Penatti and R. da Silva Torres, "Color descriptors for web image retrieval: A comparative study," *Proceedings of the Brazilian Symposium on Computer Graphics and Image Processing*, pp. 163–170, October 2008, Campo Grande, Brazil.

[158] O. A. B. Penatti, E. Valle, and R. d. S. Torres, "Comparative study of global color and texture descriptors for web image retrieval," *Journal of Visual Communication and Image Representation*, vol. 23, no. 2, pp. 359–380, February 2012.

[159] M. J. Swain and D. H. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[160] G. Pass, R. Zabih, and J. Miller, "Comparing images using color coherence vectors," *Proceedings of the 4th ACM international conference on Multimedia*, pp. 65–73, 1997, Boston, MA.

[161] G. Paschos, I. Radev, and N. Prabakar, "Image content-based retrieval using chromaticity moments," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 5, pp. 1069–1072, September 2003.

[162] A. Utenpattanant, O. Chitsobhuk, and A. Khawne, "Color descriptor for image retrieval in wavelet domain," *Proceedings of the 8th International Conference on Advanced Communication Technology (ICACT)*, vol. 1, pp. 821–824, February 2006, Phoenix Park, Ireland.

[163] S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, July 1989.

[164] B. Manjunath, J.-R. Ohm, V. Vasudevan, and A. Yamada, "Color and texture descriptors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, pp. 703–715, June 2001.

[165] K.-L. Lee and L.-H. Chen, "An efficient computation method for the texture browsing descriptor of MPEG-7," *Image and Vision Computing*, vol. 23, no. 5, pp. 479–489, May 2005.

[166] V. Risojević, S. Momić, and Z. Babić, "Gabor descriptors for aerial image classification," *Proceedings of the 10th International Conference on Adaptive and Natural Computing Algorithms - Volume Part II*, pp. 51–60, 2011, Ljubljana, Slovenia.

[167] H. G. Feichtinger and T. Strohmer, *Gabor Analysis and Algorithms: Theory and Applications*, ser. Applied and Numerical Harmonic Analysis. Birkhäuser Boston, 1998.

[168] D. Zhang and G. Lu, "Evaluation of MPEG-7 shape descriptors against other shape descriptors," *Multimedia System*, vol. 9, pp. 15–30, July 2003.

[169] C. T. Zahn and R. Z. Roskies, "Fourier Descriptors for plane closed curves," *IEEE Transactions on Computers*, vol. 21, no. 3, pp. 269–281, March 1972.

[170] E. Persoon and K. S. Fu, "Shape discrimination using Fourier Descriptors," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 7, no. 3, pp. 170–179, March 1977.

[171] Y. Zhao and S. Belkasim, "Multiresolution Fourier Descriptors for multiresolution shape analysis," *IEEE Signal Processing Letters*, vol. 19, no. 10, pp. 692–695, October 2012.

[172] F. Mokhtarian, S. Abbasi, and J. Kittler, "Efficient and robust retrieval by shape content through curvature scale space," *Proceedings of the International Workshop on Image Databases and Multimedia Search*, pp. 35–42, 1996, Amsterdam, Netherlands.

[173] A. Dyana and S. Das, "MST-CSS (Multi-Spectro-Temporal Curvature Scale Space), a novel spatio-temporal representation for content-based video retrieval," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 8, pp. 1080–1094, August 2010.

[174] M.-K. Hu, "Visual pattern recognition by moment invariants," *IRE Transactions on Information Theory*, vol. 8, no. 2, pp. 179–187, February 1962.

[175] D. Xu and H. Li, "Geometric moment invariants," *Pattern Recognition*, vol. 41, no. 1, pp. 240–249, 2008.

[176] M. R. Teague, "Image analysis via the general theory of moments," *Journal of the Optical Society of America*, vol. 70, no. 8, pp. 920–930, August 1980.

[177] S. Li, M.-C. Lee, and C.-M. Pun, "Complex zernike moments features for shape-based image retrieval," *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, vol. 39, no. 1, pp. 227–237, January 2009.

[178] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, October 2005.

[179] N. Pinto, Y. Barhomi, D. Cox, and J. DiCarlo, "Comparing state-of-the-art visual features on invariant object recognition tasks," *Proceedings of the IEEE Workshop on Applications of Computer Vision (WACV)*, pp. 463–470, January 2011, Kona, HI.

[180] A. Bosch, A. Zisserman, and X. Muoz, "Image classification using random forests and ferns," *Proceedings of the IEEE 11th International Conference on Computer Vision (ICCV)*, pp. 1–8, October 2007, Rio de Janeiro, Brazil.

[181] Z. Chen, F. Yang, A. Lindner, G. Barrenetxea, and M. Vetterli, "How is the weather: Automatic inference from images," *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pp. 1853–1856, September 2012, Orlando, FL.

[182] T. Kanungo, D. Mount, N. Netanyahu, C. Piatko, R. Silverman, and A. Wu, "An efficient k-means clustering algorithm: analysis and implementation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 881–892, July 2002.

[183] E. N. Mortensen, H. Deng, and L. Shapiro, "A SIFT descriptor with global context," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 184–190, 2005, San Diego, CA.

[184] S. Belongie, J. Malik, and J. Puzicha, "Shape context: A new descriptor for shape matching and object recognition," *Proceedings of the Neural Information Processing Systems Conference*, pp. 831–837, 2000, Denver, CO.

[185] S. Belongie, J. Malik and J. Puzicha, "Shape matching and object recognition using Shape Contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 509–522, April 2002.

[186] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Computing Surveys*, vol. 40, no. 2, pp. 5:1–5:60, May 2008, New York, NY.

[187] N. Singhai and S. K. Shandilya, "A survey on: content based image retrieval systems," *International Journal of Computer Applications*, vol. 2, no. 4, pp. 22–26, 2010.

[188] M. Jain and S. Singh, "A survey on: Content based image retrieval systems using clustering techniques for large data sets," *International Journal of Managing Information Technology*, vol. 3, no. 4, pp. 23–29, 2011.

[189] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, August 2000.

[190] F. Zhu, M. Bosch, N. Khanna, C. Boushey, and E. Delp, "Multilevel segmentation for food classification in dietary assessment," *Proceedings of 7th International Symposium on Image and Signal Processing and Analysis*, pp. 337–342, September 2008, Dubrovnik, Croatia.

[191] D. Ilea and P. Whelan, "CTex - an adaptive unsupervised segmentation algorithm based on color-texture coherence," *IEEE Transactions on Image Processing*, vol. 17, no. 10, pp. 1926–1939, October 2008.

[192] J. Li and J. W., "Studying digital imagery of ancient paintings by mixtures of stochastic models," *IEEE Transactions on Image Processing*, vol. 13, no. 3, pp. 340–353, March 2004.

[193] H. Muller, T. Pun, and D. Squire, "Learning from user behavior in image retrieval: Application of market basket analysis," *International Journal of Computer Vision*, vol. 56, pp. 65–77, January 2004.

[194] J. He, H. Tong, M. Li, H.-J. Zhang, and C. Zhang, "Mean version space: a new active learning method for content-based image retrieval," *Proceedings of the ACM SIGMM International Workshop on Multimedia Information Retrieval*, pp. 15–22, October 2004, New York, NY.

[195] F. Jing, M. Li, H.-J. Zhang, and B. Zhang, "An efficient and effective region-based image retrieval framework," *IEEE Transactions on Image Processing*, vol. 13, no. 5, pp. 699–709, May 2004.

[196] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2161–2168, June 2006, Washington, DC.

[197] S. Tong and E. Chang, "Support vector machine active learning for image retrieval," *Proceedings of the ACM international conference on Multimedia*, pp. 107–118, October 2001, Ottawa, Canada.

[198] Z. Su, H. Zhang, S. Li, and S. Ma, "Relevance feedback in content-based image retrieval: Bayesian framework, feature subspaces, and progressive learning," *IEEE Transactions on Image Processing*, vol. 12, no. 8, pp. 924–937, August 2003.

[199] Y. Wu, Q. Tian, and T. Huang, "Discriminant-EM algorithm with application to image retrieval," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 222–227, June 2000, Hilton Head Island, NC.

[200] X. He, O. King, W.-Y. Ma, M. Li, and H.-J. Zhang, "Learning a semantic space from user's relevance feedback for image retrieval," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 1, pp. 39–48, January 2003.

[201] F. Jing, M. Li, H.-J. Zhang, and B. Zhang, "Relevance feedback in region-based image retrieval," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 5, pp. 672–681, May 2004.

[202] X. S. Zhou and T. S. Huang, "Relevance feedback in image retrieval: A comprehensive review," *Multimedia Systems*, vol. 8, pp. 536–544, April 2003.

[203] A. Jaimes, K. Omura, T. Nagamine, and K. Hirata, "Memory cues for meeting video retrieval," *Proceedings of the the 1st ACM Workshop on Continuous Archival and Retrieval of Personal Experiences*, pp. 74–85, October 2004, New York, NY.

[204] C. Yang, J. Yang, and D. Feng, "Magazine image retrieval with camera-phone," *Lecture Notes in Electrical Engineering, Recent Progress in Data Engineering and Internet Technology*, vol. 156, pp. 55–60, 2013.

[205] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "Sketch-based image retrieval: Benchmark and bag-of-features descriptors," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 11, pp. 1624–1636, November 2011.

[206] J. M. Saavedra and B. Bustos, "Sketch-based image retrieval using keyshapes," *Multimedia Tools and Applications*, pp. 1–30, September 2013.

[207] A. Del Bimbo and P. Pala, "Visual image retrieval by elastic matching of user sketches," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 2, pp. 121–132, February 1997.

[208] A. Chalechale, G. Naghdy, and A. Mertins, "Sketch-based image matching using angular partitioning," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 35, no. 1, pp. 28–41, January 2005.

[209] D.-C. Tseng, Y.-F. Li, and C.-T. Tung, "Circular histogram thresholding for color image segmentation," *Proceedings of the 3rd International Conference on Document Analysis and Recognition*, vol. 2, pp. 673–676, August 1995, Montreal, Canada.

[210] D.-C. Tseng and C.-H. Chang, "Color segmentation using perceptual attributes," *Proceedings of the 11th IAPR International Conference on Pattern Recognition*, vol. 3, pp. 228–231, September 1992, La Haye, Holland.

[211] J. Brand and J. Mason, "Skin probability map and its use in face detection," *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, vol. 1, pp. 1034–1037, October 2001, Thessaloniki, Greece.

[212] Z. Xue, D. Shen, and S. Wong, "Tissue probability map constrained CLASSIC for increased accuracy and robustness in serial image segmentation," *Proceedings of the SPIE Symposium on Medical Imaging*, vol. 7258, pp. 725 904–1–9, February 2009, Lake Buena Vista, FL.

[213] J. Jiang, Y. Zhao, and S.-G. Wang, "Color correction of smartphone photos with prior knowledge," *Proceedings of the IS&T/SPIE Electronic Imaging on Imaging and Printing in a Web 2.0 World III*, vol. 8302, pp. 83 020H:1–83 020H:6, January 2012, Burlingame, CA.

[214] R. M. Boynton, *Human Color Vision*. Holt Rinehart and Winston, 1979.

[215] E. Schubert, *Light-emitting Diodes*. Cambridge University Press, 2003, ch. Human eye sensitivity and photometric quantities, pp. 275–291.

[216] J. Chae, I. Woo, S. Kim, R. Maciejewski, F. Zhu, E. Delp, C. Boushey, and D. Ebert, "Volume estimation using food specific shape templates in mobile image-based dietary assessment," *Proceedings of the IS&T/SPIE Conference on Computational Imaging IX*, vol. 7873, pp. 78 730K–1–78 730K–8, January 2011, San Francisco, CA.

[217] M. Rufli, D. Scaramuzza, and R. Siegwart, "Automatic detection of checkerboards on blurred and distorted images," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and System*, pp. 3121–3126, September 2008, Nice, France.

[218] J. J. McCann, "Color spaces for color-gamut mapping," *Journal of Electronic Imaging*, vol. 8, no. 4, pp. 354–364, October 1999.

[219] G. Wyszecki and W. S. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae. Second Edition*, ser. Wiley Series in Pure and Applied Optics. Wiley, John, and Sons, New York, N.Y., 1982.

[220] F. López, J. Valiente, R. Baldrich, and M. Vanrell, "Fast surface grading using color statistics in the CIELab space," *Proceedings of the Second Iberian Conference on Pattern Recognition and Image Analysis (IBPRIA)*, pp. 666–673, June 2005, Storil, Portugal.

[221] *Recommendation ITU-R BT.709, Parameter values for the HDTV standards for production and international programme exchange*, International Telecommunications Union, Geneva, Switzerland, 1990.

[222] G. Strang, *Introduction to Applied Mathematics*. Wellesley-Cambridge Press, 1986.

[223] C. Poynton, *Digital Video and HDTV Algorithms and Interfaces*, 1st ed. San Francisco, CA: Morgan Kaufmann Publishers Inc., 2003.

[224] L. Lam, S. W. Lee, and C. Y. Suen, "Thinning methodologies - A comprehensive survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 9, pp. 869–885, September 1992.

[225] Z. Guo and R. W. Hall, "Parallel thinning with two-subiteration algorithms," *Communications of the ACM*, vol. 32, no. 3, pp. 359–373, March 1989.

[226] P. V. C. Hough, "Machine analysis of bubble chamber pictures," *Proceedings of the International Conference on High Energy Accelerators and Instrumentation*, pp. 554–558, September 1959, Geneva, Switzerland.

[227] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, January 1972.

[228] J. E. Bresenham, "Algorithm for computer control of a digital plotter," *IBM Systems Journal*, vol. 4, no. 1, pp. 25–30, 1965.

[229] E. Hancer and R. Samet, "Advanced contour reconnection in scanned topographic maps," *Proceedings of the 5th International Conference on Application of Information and Communication Technologies (AICT)*, pp. 1–5, October 2011, Baku, Azerbaijan.

[230] W. Wang, H. Pottmann, and Y. Liu, "Fitting B-spline curves to point clouds by curvature-based squared distance minimization," *ACM Transactions on Graphics*, vol. 25, no. 2, pp. 214–238, April 2006.

[231] D. G. Lowe, "Object recognition from local scale-invariant features," *Proceedings of the International Conference on Computer Vision*, vol. 2, pp. 1150–1157, September 1999, Kerkyra, Greece.

[232] P. Perona, "Deformable kernels for early vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 5, pp. 488–499, May 1995.

[233] E. Tola, V. Lepetit, and P. Fua, "DAISY: An efficient dense descriptor applied to wide baseline stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 5, pp. 815–830, May 2010.

[234] T.-S. Chen, T.-H. Tsai, Y.-T. Chen, C.-C. Lin, R.-C. Chen, S.-Y. Li, and H.-Y. Chen, "A combined k-means and hierarchical clustering method for improving the clustering efficiency of microarray," *Proceedings of International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, pp. 405–408, December 2005, Hong Kong, China.

[235] T. Su and J. Dy, "A deterministic method for initializing k-means clustering," *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 784–786, November 2004, Boca Raton, FL.

[236] R. Xu and D. Wunsch II, "Survey of clustering algorithms," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 645–678, May 2005.

[237] R. Bellman, *Adaptive Control Processes: A Guided Tour*. Princeton University Press, 1961.

[238] K. S. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, "When is "nearest neighbor" meaningful?" *Proceedings of the 7th International Conference on Database Theory*, pp. 217–235, 1999, London, United Kingdom.

[239] S. Arya, D. M. Mount, and O. Narayan, "Accounting for boundary effects in nearest-neighbor searching," *Discrete & Computational Geometry*, vol. 16, no. 2, pp. 155–176, 1996.

[240] S. Berchtold, C. Böhm, D. A. Keim, and H.-P. Kriegel, "A cost model for nearest neighbor search in high-dimensional data space," *Proceedings of the 16th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pp. 78–86, 1997, Tucson, AR.

[241] D. Coomans and D. L. Massart, "Alternative k-nearest neighbour rules in supervised pattern recognition: Part 1. k-nearest neighbour classification by using alternative voting rules," *Analytica Chimica Acta*, vol. 136, no. 0, pp. 15–27, 1982.

[242] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, June 2009, Miami, FL.

[243] X. Wang, M. Yang, T. Cour, S. Zhu, K. Yu, and T. X. Han, "Contextual weighting for vocabulary tree based image retrieval," *Proceedings of the 2011 International Conference on Computer Vision*, pp. 209–216, 2011, Washington, DC.

[244] B. Momjian, *PostgreSQL: Introduction and Concepts*. Boston, MA: Addison-Wesley Longman Publishing Co., Inc., 2001.

[245] "Android Developers. Platform Versions as of February 4, 2014." `developer.android.com/about/dashboards`

[246] E. Lafortune, "ProGuard: Optimizer and obfuscator in the Android SDK," 2006. `proguard.sourceforge.net`

[247] J. D. Touch, "Performance analysis of MD5," *ACM SIGCOMM Computer Communication Review*, pp. 77–86, October 1995.

[248] C. Grigorescu and N. Petkov, "Distance sets for shape filters and shape recognition," *IEEE Transactions on Image Processing*, vol. 12, no. 10, pp. 1274–1286, October 2003.

[249] T. Gevers and A. W. M. Smeulders, "Color-based object recognition," *Pattern Recognition*, pp. 453–464, March 1999.

[250] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, November 1998.

[251] G. Loy and N. Barnes, "Fast shape-based road sign detection for a driver assistance system," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, pp. 70–75, September 2004, Stockholm, Sweden.

[252] S. Maldonado-Bascon, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gomez-Moreno, and F. Lopez-Ferreras, "Road-sign detection and recognition based on support vector machines," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 2, pp. 264–278, June 2007.

[253] R. Malik, J. Khurshid, and S. Ahmad, "Road sign detection and recognition using colour segmentation, shape analysis and template matching," *Proceedings of the International Conference on Machine Learning and Cybernetics*, vol. 6, pp. 3556–3560, August 2007, Hong Kong, China.

[254] J. Greenhalgh and M. Mirmehdi, "Real-time detection and recognition of road traffic signs," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1498–1506, December 2012.

[255] O. R. Mitchell and T. A. Grogan, "Global and partial shape discrimination for computer vision," *Optical Engineering*, vol. 23, no. 5, pp. 484–491, October 1984.

[256] R. C. Gonzalez, *Digital Image Processing*, 2nd ed. New Jersey: Prentice Hall, 2000.

[257] F. Larsson, M. Felsberg, and P.-E. Forssen, "Correlating Fourier Descriptors of local patches for road sign recognition," *IET Computer Vision*, vol. 5, pp. 244–254, January 2011.

[258] P. van Otterloo, *A Contour-Oriented Approach to Shape Analysis*, 2nd ed. Englewood Cliffs, New Jersey: Prentice-Hall International, 2000.

[259] R. Chellappa and R. Bagdazian, "Fourier coding of image boundaries," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 1, pp. 102–105, January 1984.

[260] C. Singh and P. Sharma, "Performance analysis of various local and global shape descriptors for image retrieval," *Multimedia Systems*, vol. 19, no. 4, pp. 339–357, July 2013.

[261] I. Kunttu, L. Lepisto, J. Rauhumaa, and A. Visa, "Multiscale Fourier Descriptor for shape-based image retrieval," *Proceedings of the IEEE Conference on Pattern Recognition*, pp. 765–768, August 2004, Cambridge, United Kingdom.

[262] N. M. Tahir, A. Hussain, and M. M. Mustafa, "Fourier Descriptor for pedestrian shape recognition using support vector machine," *Proceedings of the IEEE International Symposium on Signal Processing and Information*, pp. 636–641, December 2007, Cairo, Egypt.

[263] O. van Kaick, G. Hamarneh, H. Zhang, and P. Wighton, "Contour correspondence via ant colony optimization," *Proceedings of the Pacific Conference on Computer Graphics and Applications*, pp. 271–280, October 2007, Maui, HI.

[264] M. Jie, Z. Zhiwei, T. HongMei, and Z. QuanMing, "Fast Fourier Descriptor method of the shape feature in low resolution images," *Proceedings of the IEEE Conference Wireless Communications Networking and Mobile Computing*, pp. 1–4, September 2010, Chengdu, China.

[265] A. Broggi, P. Cerri, P. Medici, P. Porta, and G. Ghisio, "Real time road signs recognition," *IEEE Intelligent Vehicles Symposium*, pp. 981–986, June 2007, Istambul, Turkey.

[266] L. Song and Z. Liu, "Color-based traffic sign detection," *International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering*, pp. 353–357, June 2012, Chengdu, China.

[267] A. Borji and L. Itti, "State-of-the-art in visual attention modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 185–207, January 2013.

[268] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," *Proceedings of the Annual Conference on Neural Information Processing Systems*, pp. 545–552, December 2006, Vancouver, Canada.

[269] X. Hou and L. Zhang, "Dynamic visual attention: Searching for coding length increments," *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, pp. 681–688, December 2008, Vancouver, Canada.

[270] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Global contrast based salient region detection," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 409–416, June 2011, Colorado Springs, CO.

[271] C. Kim and P. Milanfar, "Visual saliency in noisy images," *Journal of Vision*, vol. 13, no. 4, pp. 1–14, March 2013.

[272] X. Hou, J. Harel, and C. Koch, "Image signature: Highlighting sparse salient regions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 194–201, January 2012.

[273] J. Li, M. D. Levine, X. An, X. Xu, and H. He, "Visual saliency based on scale-space analysis in the frequency domain," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 4, pp. 996–1010, April 2013.

[274] R. Belaroussi, P. Foucher, J.-P. Tarel, B. Soheilian, P. Charbonnier, and N. Paparoditis, "Road sign detection in images: A case study," *Proceedings of the International Conference on Pattern Recognition*, pp. 484–488, August 2010, Istambul, Turkey.

[275] A. Mogelmose, M. Trivedi, and T. Moeslund, "Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1484–1497, December 2012.

[276] K. L. Bouman, G. Abdollahian, M. Boutin, and E. J. Delp, "A low complexity sign detection and text localization method for mobile applications," *IEEE Transactions on Multimedia*, vol. 13, no. 5, pp. 922–934, October 2011.

[277] D. Pao, H. Li, and R. Jayakumar, "Shapes recognition using the straight line Hough transform: theory and generalization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 11, pp. 1076–1089, November 1992.

[278] S. Houben, "A single target voting scheme for traffic sign detection," *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 124–129, June 2011, Baden-Baden, Germany.

[279] H. Fleyeh and P. Zhao, "A contour-based separation of vertically attached traffic signs," *Proceedings of the Annual Conference of Industrial Electronics*, pp. 1811–1816, November 2008, Orlando, FL.

[280] L.-W. Tsai, J.-W. Hsieh, C.-H. Chuang, Y.-J. Tseng, K.-C. Fan, and C.-C. Lee, "Road sign detection using eigen colour," *IET Computer Vision*, no. 3, pp. 164–177, September 2008.

[281] G. Loy and A. Zelinsky, "Fast radial symmetry for detecting points of interest," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 959–973, August 2003.

[282] N. Barnes, A. Zelinsky, and L. Fletcher, "Real-time speed sign detection using the radial symmetry detector," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 2, pp. 322–332, June 2008.

[283] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, May 2004.

[284] C. Keller, C. Sprunk, C. Bahlmann, J. Giebel, and G. Baratoff, "Real-time recognition of U.S. speed signs," *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 518–523, June 2008, Eindhoven, Netherlands.

[285] X. Baro, S. Escalera, J. Vitria, O. Pujol, and P. Radeva, "Traffic sign recognition using evolutionary AdaBoost detection and Forest-ECOC classification," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 1, pp. 113–126, March 2009.

[286] A. Rostampour and P. Madhvapathy, "Shape recognition using simple measures of projections," *Proceedings of the Annual International Phoenix Conference on Computers and Communications*, pp. 474–479, March 1988, Scottsdale, AR.

[287] P. Gil-Jimenez, S. Lafuente-Arroyo, H. Gomez-Moreno, F. Lopez-Ferreras, and S. Maldonado-Bascon, "Traffic sign shape classification evaluation. part II. FFT applied to the signature of blobs," *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 607–612, June 2005, Las Vegas, NV.

[288] A. W. Haddad, S. Huang, M. Boutin, and E. J. Delp, "Detection of symmetric shapes on a mobile device with applications to automatic sign interpretation," *Proceedings of the IS&T/SPIE Electronic Imaging on Multimedia on Mobile Devices*, vol. 8304, January 2012, San Francisco, CA.

[289] "Ocrad - GNU Project - Free Software Foundation (FSF)." `www.gnu.org/software/ocrad`

[290] A. Parra, A. W. Haddad, M. Boutin, and E. Delp, "A method for translating printed documents using a hand-held device," *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, July 2011, Barcelona, Spain.

[291] X. Hou, J. Harel, and C. Koch, "Image signature: Highlighting sparse salient regions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 194–201, January 2012.

[292] S. Suzuki and K. Abe, "Topological structural analysis of digitized binary images by border following," *Computer Vision, Graphics, and Image Processing*, vol. 30, no. 1, pp. 32–46, April 1985.

[293] J. Sklansky, "Finding the convex hull of a simple polygon," *Pattern Recognition Letters*, vol. 1, no. 2, pp. 79–83, December 1982.

[294] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, January 1979.

[295] C. Correa, C. Valero, and P. Barreiro, "Row crop's identification through Hough transform using images segmented by robust fuzzy possibilistic c-means," *Proceedings of the Spanish Association for Artificial Intelligence*, November 2011, La Laguna, Spain.

[296] H.-D. Cheng and Y. Sun, "A hierarchical approach to color image segmentation using homogeneity," *IEEE Transactions on Image Processing*, vol. 9, no. 12, pp. 2071–2082, 2000.

[297] P. Soille, *Morphological Image Analysis: Principles and Applications*. Springer-Verlag, 1999.

[298] G. Anelli, A. Broggi, and G. Destri, "Decomposition of arbitararily-shaped morphological structuring elements using genetic algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 2, pp. 217–224, 1998.

[299] H. Park and R. Chin, "Decomposition of arbitararily-shaped morphological structuring elements," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 1, pp. 2–15, 1995.

[300] R. F. Gonzalez, R. E. Woods, and S. L. Eddins, *Digital Image Processing Using MATLAB*. Upper Saddle River, NJ: Prentice-Hall, Inc., 2003.

[301] S. Pereira and T. Pun, "Robust template matching for affine resistant image watermarks," *IEEE Transactions on Image Processing*, vol. 9, no. 6, pp. 1123–1129, 2000.

[302] F. Essannouni and D. Aboutajdine, "Fast frequency template matching using higher order statistics," *IEEE Transactions on Image Processing*, vol. 19, no. 3, pp. 826–830, 2010.

[303] I. Bartolini, P. Ciaccia, and M. Patella, "WARP: Accurate retrieval of shapes using phase of fourier descriptors and time warping distance." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 1, pp. 142–147, 2005.

[304] R. P. Brent, "Fast multiple-precision evaluation of elementary functions," *Journal of the ACM*, vol. 23, pp. 242–251, April 1976.

[305] D. Pascale, "RGB coordinates of the Macbeth ColorChecker," *The BabelColor Company*, pp. 1–15, June 2006, Montreal, Canada.

[306] S. V. Stehman, "Selecting and interpreting measures of thematic classification accuracy," *Remote Sensing of Environment*, vol. 62, no. 1, pp. 77–89, 1997.

[307] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing and Management*, vol. 45, no. 4, pp. 427–437, 2009.

[308] E. M. Voorhees, "Variations in relevance judgments and the measurement of retrieval effectiveness," *Information Processing & Management*, vol. 36, no. 5, pp. 697–716, 2000.

[309] K. E. A. Van de Sande, T. Gevers, and C. G. M. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1582–1596, September 2010.

[310] J. Huang, H. Liu, J. Shen, and S. Yan, "Towards efficient sparse coding for scalable image annotation," *Proceedings of the 21st ACM International Conference on Multimedia*, pp. 947–956, October 2013, Barcelona, Spain.

[311] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with Stroke Width Transform," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2963–2970, June 2010, San Francisco, CA.

[312] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, November 1998.

[313] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, pp. 545–552, December 2006, Vancouver, Canada.

[314] B. Zhao, A. Parra, and E. J. Delp, "Mobile-based hazmat sign detection system," *Proceedings of the IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 735–738, December 2013, Austin, TX.

[315] V. Chandrasekhar, G. Takacs, D. Chen, S. Tsai, R. Grzeszczuk, and B. Girod, "CHoG: Compressed histogram of gradients: A low bit-rate feature descriptor," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2504–2511, June 2009, Miami, FL.

[316] K. Kumar, J. Liu, Y.-H. Lu, and B. Bhargava, "A survey of computation offloading for mobile systems," *Mobile Networks and Applications*, vol. 18, no. 1, pp. 129–140, February 2013.

[317] K. Mikolajczyk, B. Leibe, and B. Schiele, "Multiple object class detection with a generative model," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 26–36, June 2006, New York, NY.

[318] Y. Yang, S. Hallman, D. Ramanan, and C. Fowlkes, "Layered object detection for multi-class segmentation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3113–3120, June 2010, San Francisco, CA.

[319] M. Asmare, V. Asirvadam, and L. Iznita, "Color space selection for color image enhancement applications," *Proceedings of the International Conference on Signal Acquisition and Processing*, pp. 208–212, April 2009, Kuala Lumpur, Malaysia.

[320] M. Tkalcic and J. Tasic, "Colour spaces: Perceptual, historical and applicational background," *Proceedings of the IEEE Region 8 Eurocon 2003: Computer as a Tool*, vol. 1, pp. 304–308, September 2003, Ljubljana, Slovenia.

[321] G. H. Joblove and D. Greenberg, "Color spaces for computer graphics," *ACM SIGGRAPH Computer Graphics*, vol. 2, no. 3, pp. 20–25, August 1978.

[322] A. R. Smith, "Color gamut transform pairs," *Proceedings of the 5th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 12–19, 1978, New York, NY.

[323] A. Hanbury, "A 3D-polar coordinate colour representation well adapted to image analysis," *Proceedings of the 13th Scandinavian Conference on Image Analysis*, pp. 804–811, June-July 2003, Halmstad, Sweden.

[324] J. D. Foley and A. Van Dam, *Fundamentals of Interactive Computer Graphics*. Boston, MA: Addison-Wesley Longman Publishing Co., Inc., 1982.

[325] M. Agoston, *Computer Graphics and Geometric Modeling: Implementation and Algorithms*. Springer, 2005.

[326] *Recommendation ITU-R BT.601, Encoding Parameters of Digital Television for Studios*, International Telecommunications Union, Geneva, Switzerland, 1992.

[327] J. D. Foley, A. van Dam, S. Feiner, and J. Hughes, *Computer Graphics: Principles and Practice*, 2nd ed. Addison-Wesley, 1990.

APPENDICES

A. RGB TO Y'CH COLOR SPACE CONVERSION

An image captured using our Android application is saved as 32-bit RGB JPEG file, where each pixel is a packed 32-bit integer containing the alpha, R, G and B color components from most to least significant bits respectively. Note that a JPEG image does not have an alpha channel and it is automatically set to zero by the Android bitmap Application Programming Interface (API). From these packed RGB bits we create a three-dimensional array to store the R, G and B components in their unpacked bit representations.

The RGB color space is psychologically non-intuitive because humans have problems with the visualization of a color defined in RGB [319]. The attributes of hue and saturation are the most natural way for humans to perceive colors [320]. The separation of the luma component from the chrominance information is advantageous in image processing. Therefore, we chose to transform the pixels in the image from the RGB color space to our new HSL-based color space, which we call the Y'CH color space, where we carefully define the three dimensions as luma, chroma and hue. We choose chroma over saturation because it better represents human perception of the variation in color purity with respect to luma. In the literature, saturation is defined as relative chroma [321, 322], and the difference must be taken into consideration. For example, the HSL color space is symmetrical with respect to luma, taking the shape of a cylinder. When using chroma the cylinder gets narrower as we move from the center of the neutral axis, forming a shape similar to a bicone [321]. Note that Figures 3.8 and 3.10 illustrate the Y'CH color space solid representation as a bicone for simplicity. However, its true shape is shown in Figure A.3, where not all the primaries lie in the same plane.

We can convert from RGB to our Y'CH in many ways. In this section we describe two approaches. The first one uses just arithmetic operations, while the second also

uses trigonometric operations. We conclude in Section 5 that the first approach is asymptotically faster and hence it is the method that we implemented in our Android application described in Section 3.8.3.

Our first approach for transforming from RGB to Y'CH, which we call the *arithmetic approach*, is illustrated in Figure A.1. First, we interpret the RGB cube as being tilted so that the black and white vertices are positioned at the top and the bottom of the neutral axis (vertical axis), respectively. Second, we project the tilted cube onto a plane perpendicular to the neutral axis, thus forming a hexagon. The chroma (C) and hue (H) components in our model are defined with respect to this hexagonal projection (Figure A.2). Chroma is the distance from the origin of the hexagon to its edge. We can define it as the difference between the largest and the smallest values of an RGB triplet [323] as shown in Equation A.1. Hue is the angle that represents the angular distance from the red edge of the projection (i.e., set to zero radians) to a particular RGB projection [324, 325], as shown in Equation A.2. Note that this theoretical hue, which we define as H', is undefined for projections onto the neutral axis (i.e., $C = 0$). Also note that these definitions of chroma and hue correspond to a geometric warping of the hexagon into a circumference.

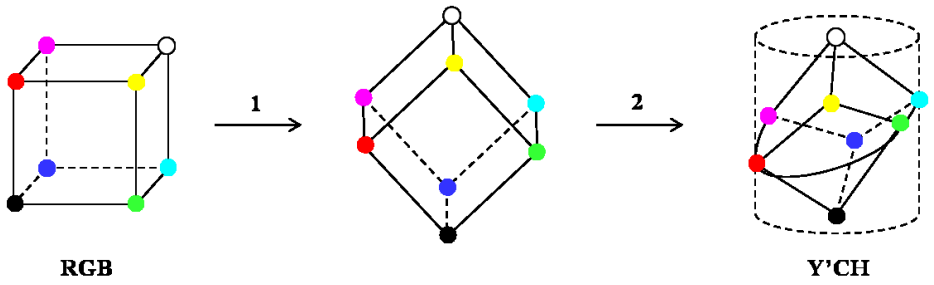


Fig. A.1.: Steps For Transforming from RGB to Y'CH Using The Arithmetic Approach.

H' is then converted to degrees, which we define as H, by multiplying by 60. This multiplication accounts for $\frac{360°}{6}$, which can be interpreted as the hexagonal analog of the unit circumference conversion from radians to degrees. That is, since 2π is the perimeter of the unit circumference, we define the conversion as $rad = \frac{360}{2\pi} \times deg$.

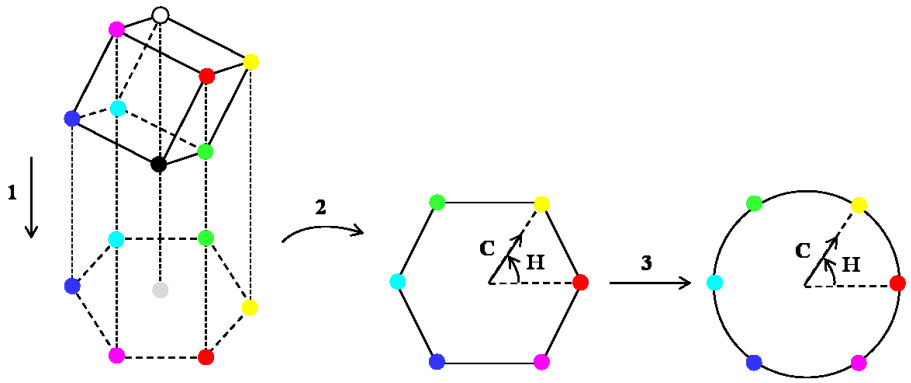


Fig. A.2.: Warping of the Hexagon Projection Into A Circumference in Our Y'CH Color Space.

Since 6 is the perimeter of the unit hexagon, we can define $rad = \frac{360}{6} \times deg = 60 \times deg$. Note that we define $H = 0$ when $C = 0$ in order to deal with the undefined hue angle for vector of magnitude zero.

Finally, our luma (Y') is the weighted average of gamma-corrected RGB color components. We define it using the Rec. 601 NTSC primaries [326], as shown in Equation A.3.

$$\begin{aligned} C &= max(R, G, B) - min(R, G, B) \\ &= M - m. \end{aligned} \tag{A.1}$$

$$H' = \begin{cases} \frac{G-B}{C} & \text{if M=R} \\ \frac{B-R}{C} + 2 & \text{if M=G} \\ \frac{R-G}{C} + 4 & \text{if M=B} \\ \text{undefined} & \text{if C=0} \end{cases} \tag{A.2}$$

$$Y = 0.299R + 0.587G + 0.114B. \tag{A.3}$$

Using these equations, our Y'CH color space is defined in $0 \leq H < 360$ (or $0 \leq H < 2\pi$ in radians), $0 \leq C \leq 1$ and $0 \leq Y \leq 1$. The resulting representation is illustrated in step 3 of Figure A.1, where each colored dot represents a fully chromatic primary. Given our definitions of luma, chroma and hue, the color space representation does not have a symmetric shape. Figure A.3 illustrates a 3D view of the Y'CH solid. Figures A.4 to A.6 illustrate different cross-sections of constant hue, where the far left and far right corners represent fully chromatic colors. Note that the primaries do not lie in a common luma plane. Also note in Figure A.5 the effect of setting $H = 0$ where $C = 0$, instead of being undefined. The neutral axis ($C = 0$) does not contain luma values, since the cross-section is not located at $H = 0$. Figure A.4, however, since it is located at $H = 0$, we do not see any discontinuity.

Figure A.7 illustrates the bottom view of our Y'CH color space representation, where the hue of different primaries can be identified.

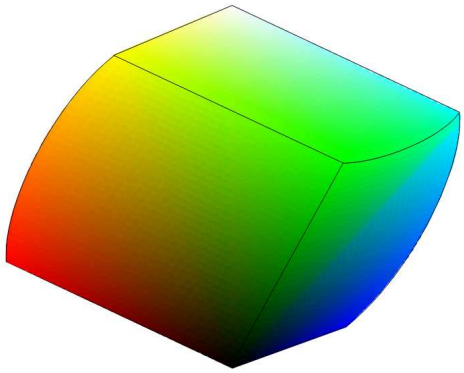


Fig. A.3.: 3D view of Our Y'CH Color Space (Using the Arithmetic Approach).

Our second approach for transforming from RGB to Y'CH, which we call the *trigonometric approach*, consists of defining the Y'CH color space using cylindrical coordinates, thus skipping the hexagon warping. First, we convert from RGB to Y'IQ using a linear transformation of the RGB cube [327], as shown in Equation A.4. With this conversion we directly obtain the Y'CH luma, which is defined again using the Rec. 601 NTSC primaries. Then, we can derive the hue and the chroma from a

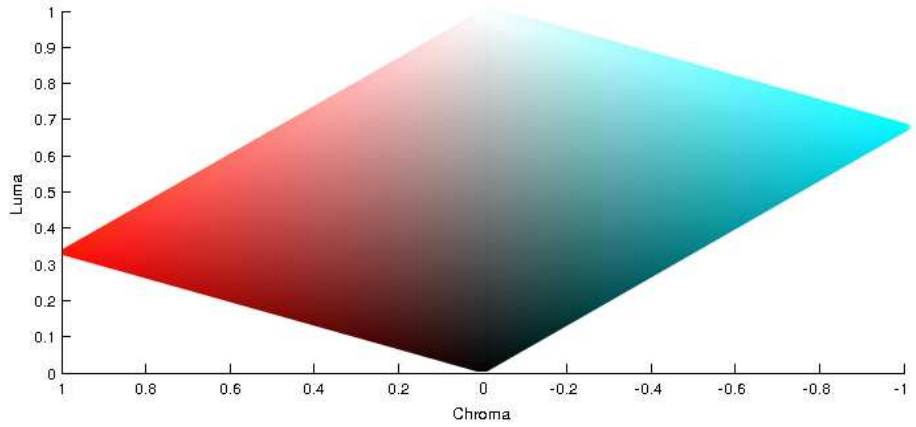


Fig. A.4.: Cross-Section of Constant Hue $H = 0$ rad in Our Y'CH Color Space.

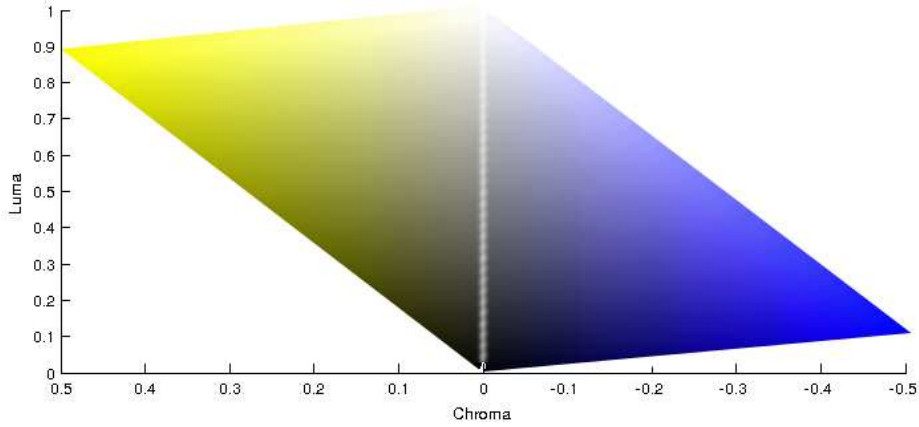


Fig. A.5.: Cross-Section of Constant Hue $H = \frac{\pi}{3}$ rad in Our Y'CH Color Space.

cylindrical transformation of I and Q [323] as shown in Equation A.6. Note that the function *atan2* in Equation A.6 is the two-argument arctangent, defined in Equation A.7.

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.595716 & -0.274453 & -0.321263 \\ 0.211456 & -0.522591 & 0.311135 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \tag{A.4}$$

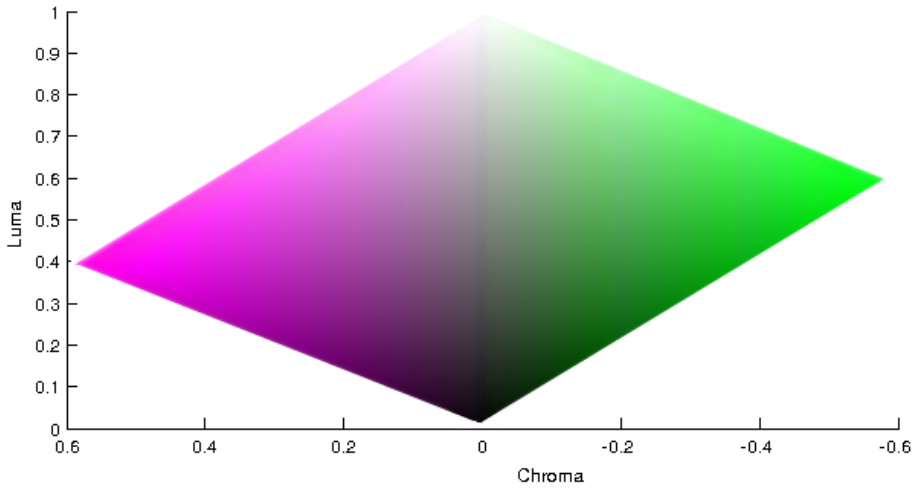


Fig. A.6.: Cross-Section of Constant Hue $H = \frac{2\pi}{3}$ rad in Our Y'CH Color Space.

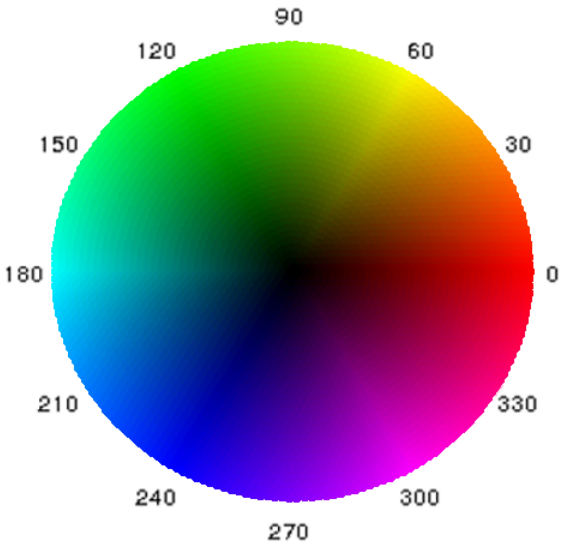


Fig. A.7.: Bottom View of Our Y'CH Color Space (Using the Arithmetic Approach).

$$H = atan2(Q, I) \tag{A.5}$$

$$C = \sqrt{I^2 + Q^2}, \tag{A.6}$$

$$atan2(I,Q) = \begin{cases} \arctan(\frac{Q}{I}) & I > 0 \\ \pi + \arctan(\frac{Q}{I}) & Q \geq 0,\ I < 0 \\ -\pi + \arctan(\frac{Q}{I}) & Q < 0,\ I < 0 \\ \frac{\pi}{2} & Q > 0,\ I = 0 \\ -\frac{\pi}{2} & Q < 0,\ I = 0 \\ \text{undefined} & Q = 0,\ I = 0 \end{cases} \tag{A.7}$$

Figure B.16 illustrates the bottom view of our Y'CH color space representation where the hue of different primaries can be identified. Note the hexagon shape.

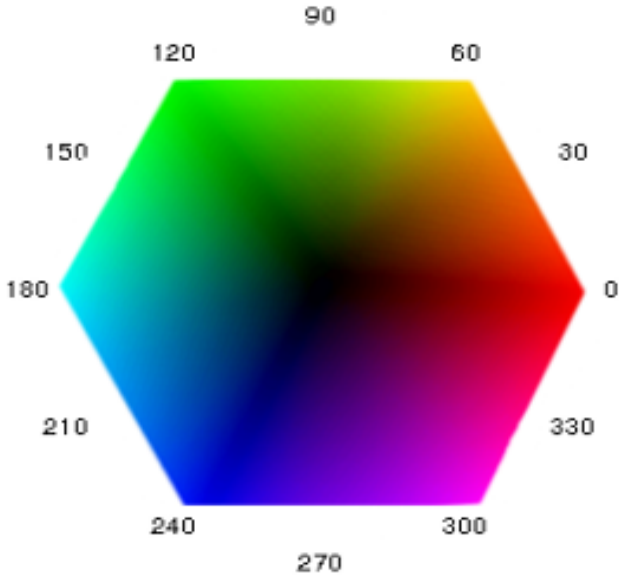


Fig. A.8.: Bottom View of Our Y'CH Color Space (Using the Trigonometric Approach).

Note that a HSL-based color space, such as Y'CH, has the disadvantage that it does not account for the complexity of the human color perception. However, since we are doing color recognition this is not an issue.

B. EXAMPLES OF GRAFFITI COLOR IMAGE SEGMENTATION

This Appendix shows examples of Color Image Segmentation Based on Gaussian Thresholding.

Fig. B.1.: Red text: $\tilde{H} = 0.49$ and $\sigma_H^2 = 0.05$.

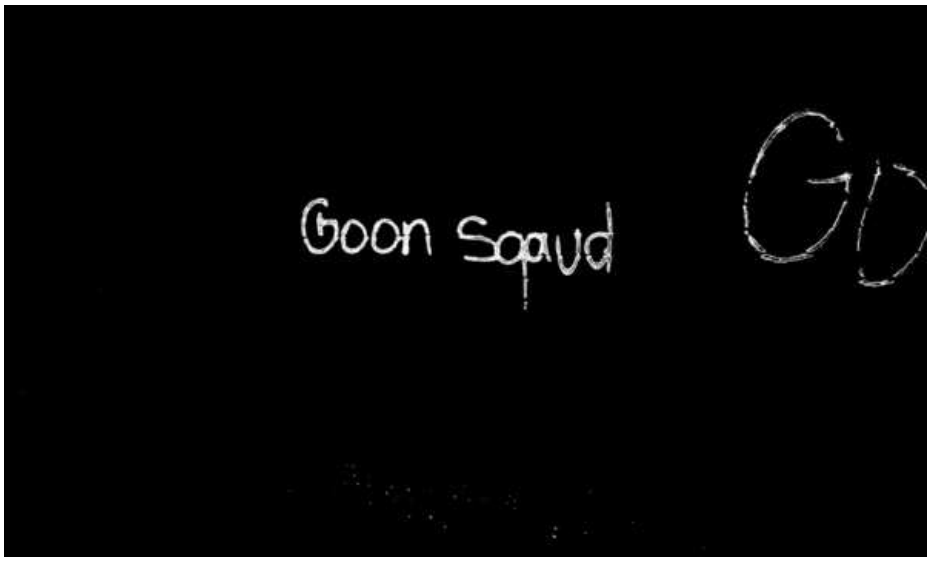


Fig. B.2.: $T_C = 0.04$.

Fig. B.3.: White text: $\tilde{Y} = 0.83$ and $\sigma^2_{\tilde{Y}} = 0.003$.

Fig. B.4.: $T_{Yb} = 0$, $T_{Yw} = 1$.

Fig. B.5.: Black text: $\tilde{Y} = 0.13$ and $\sigma^2_{\tilde{Y}} = 0.001$.

Fig. B.6.: $T_{Yb} = 0$, $T_{Yw} = 0.2$.

Fig. B.7.: Blue text: $\tilde{H} = 2.56$ and $\sigma_H^2 = 0.034$.

Fig. B.8.: $T_C = 0.04$.

Fig. B.9.: Blue text: $\tilde{H} = 2.60$ and $\sigma_H^2 = 0.020$.

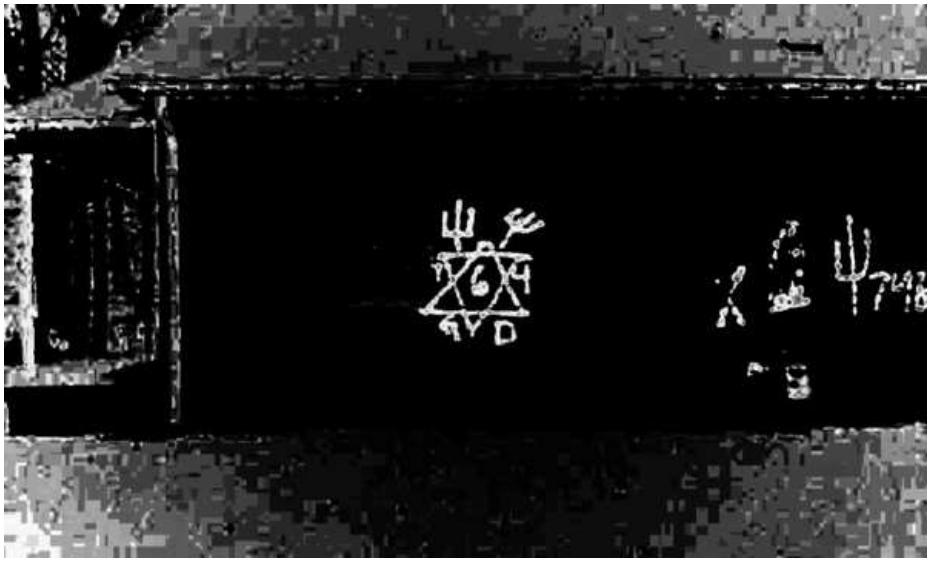


Fig. B.10.: $T_C = 0.05$.

Fig. B.11.: Blue text: $\tilde{H} = 2.73$ and $\sigma_H^2 = 0.049$.

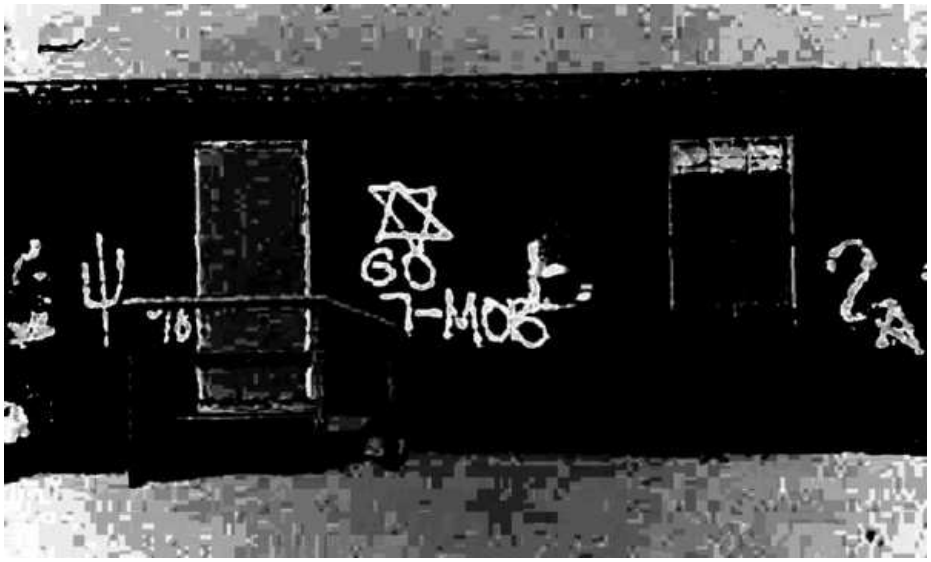


Fig. B.12.: $T_C = 0.02$.

Fig. B.13.: Black text: $\tilde{Y} = 0.17$ and $\sigma_{\tilde{Y}}^2 = 0.008$.

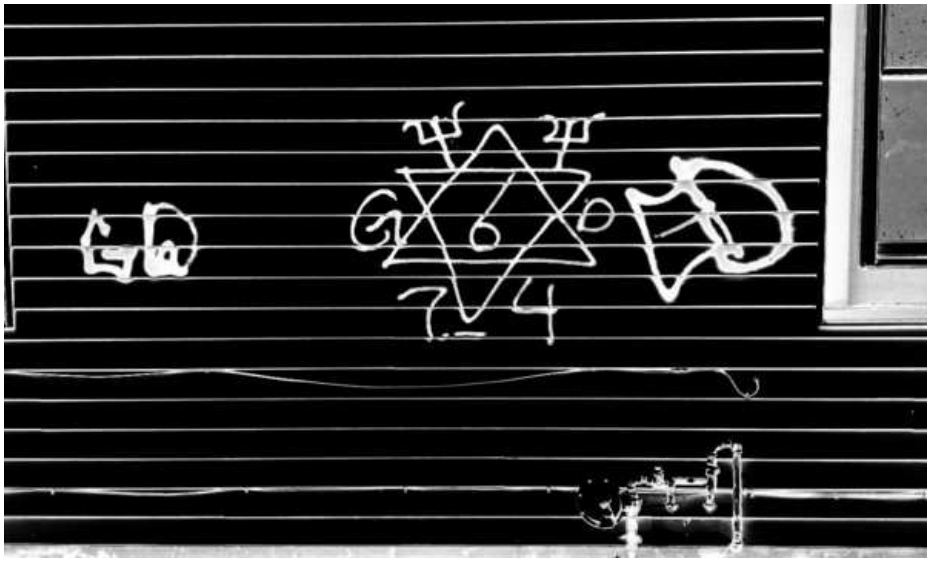


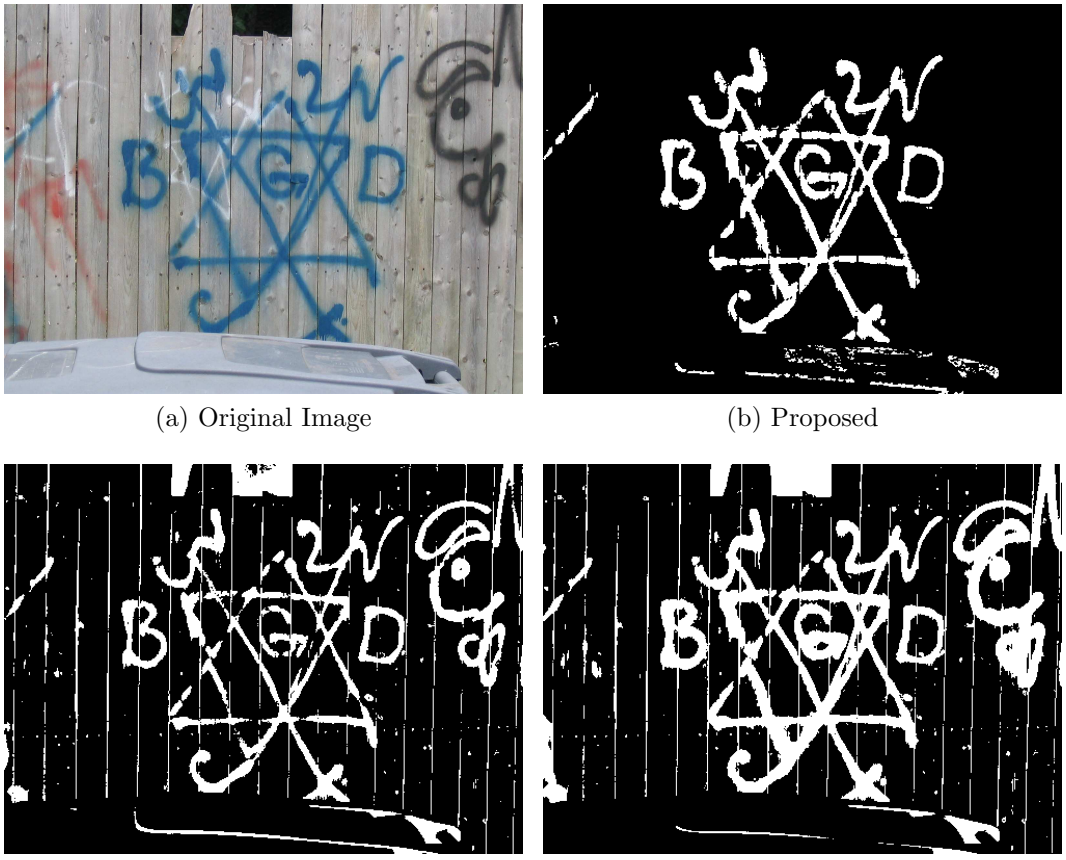
Fig. B.14.: $T_{Yb} = 0$, $T_{Yw} = 1$.

Fig. B.15.: Black text: $\tilde{Y} = 0.19$ and $\sigma_Y^2 = 0.002$.

Fig. B.16.: $T_{Yb} = 0$, $T_{Yw} = 1$.

C. IMAGE THRESHOLDING METHODS

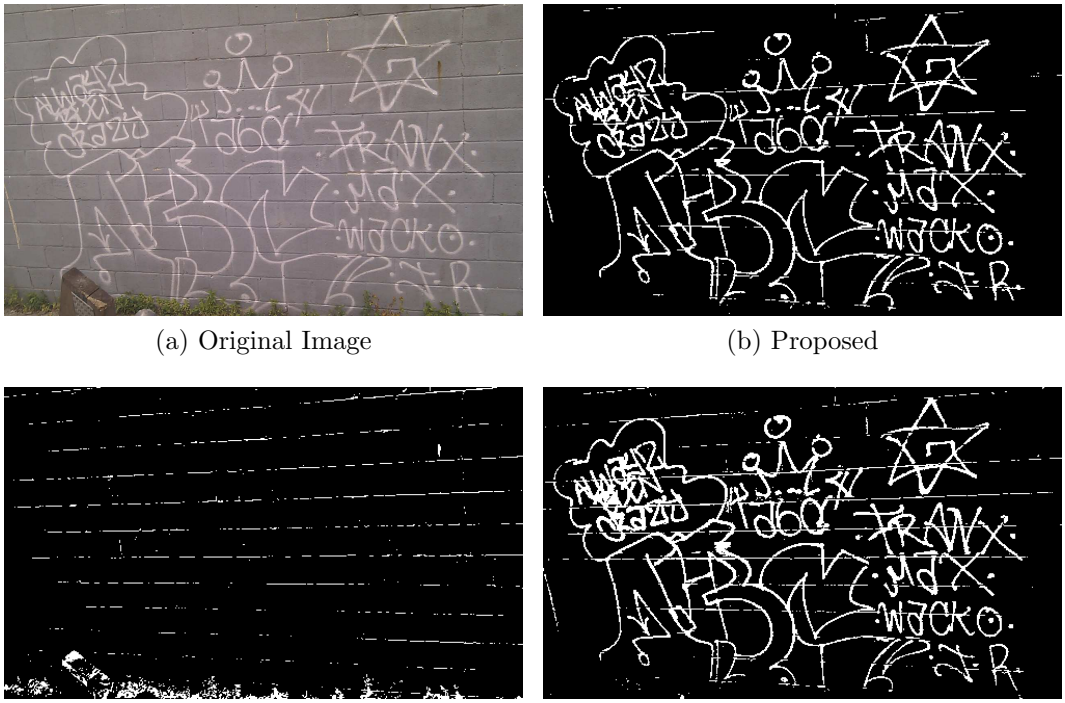
This Appendix shows the comparison of three different image thresholding methods with respect to the 20 test images used in Section 5.1.4. The thresholding methods are: 1) Our proposed combination of Color Image Segmentation Based on Gaussian Thresholding and Block-Wise Gaussian Segmentation Enhancement, 2) Niblack thresholding, 3) Otsu's method. The input of our proposed method is not just the image, but additional parameters returned from our proposed Color Recognition Based on Touchscreen Tracing (Section 3.4): boolHL indicates if the recognized color is based on hue or luma; medH is the hue median; medY is the luma median; varH is the hue variance; varY is the luma variance. The Niblack thresholding is setup with a filter radius of 25 pixels and standard deviation threshold of 0.2. Otsu's method does not need any additional configuration.



(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

Fig. C.1.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [1 3.6046, 0.3486, 0.0012, 0.0013].



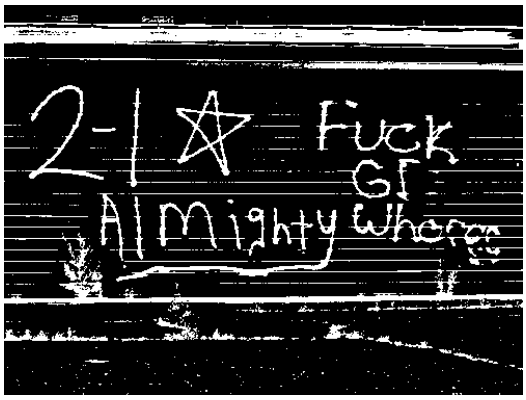
(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

Fig. C.2.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [0, 6.0868, 0.7381, 0.0075, 0.0033].

(a) Original Image

(b) Proposed



(c) Niblack

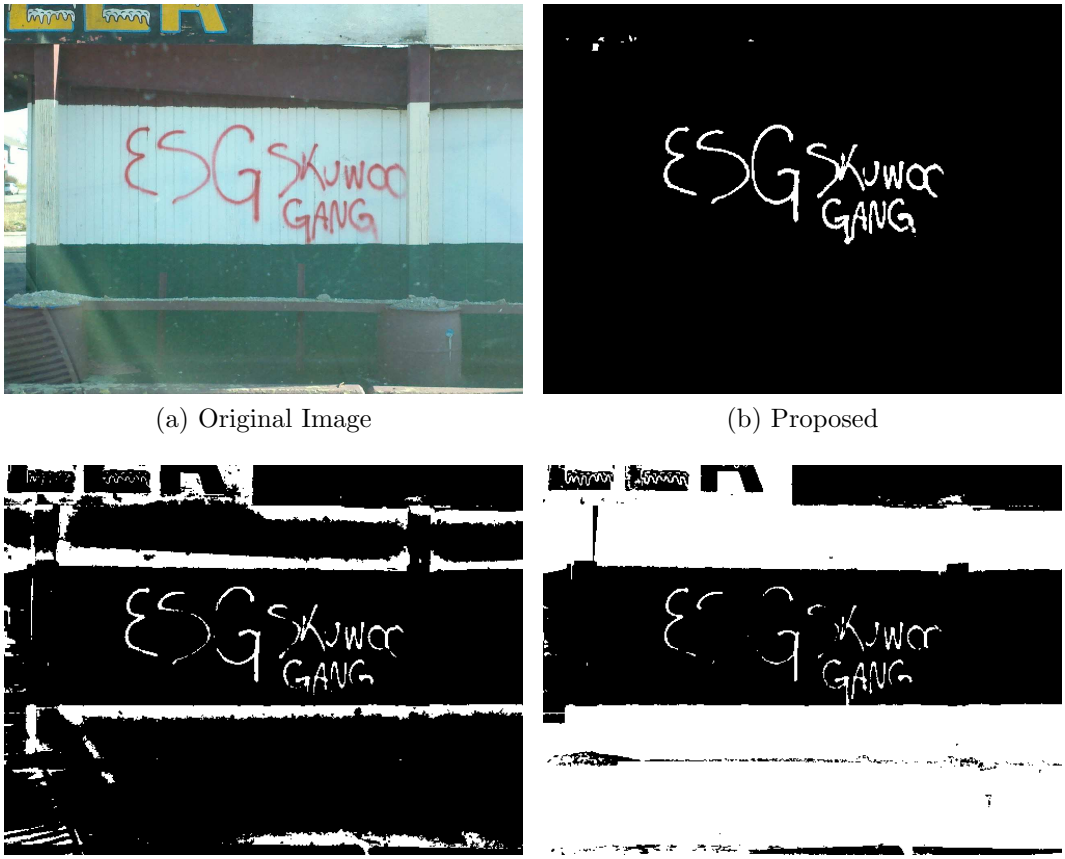
(d) Otsu

Fig. C.3.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [1, 6.0868, 0.3298, 0.0018, 0.0010].

(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

Fig. C.4.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [1, 0.2448, 0.3145, 0.0107, 0.0023].



(a) Original Image (b) Proposed

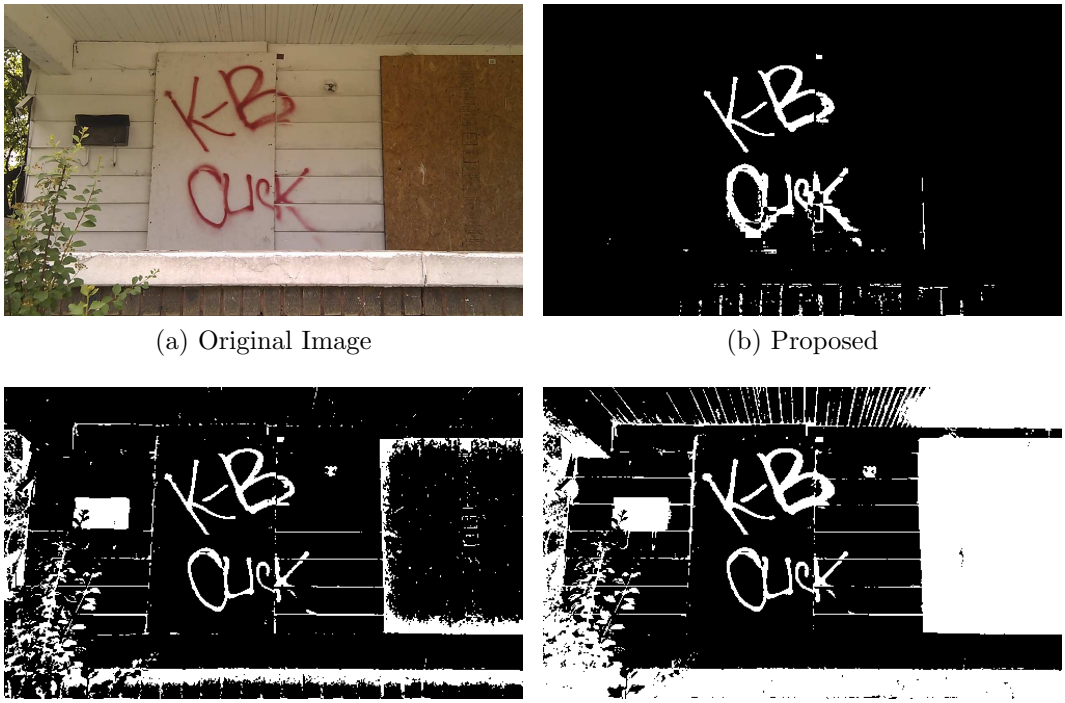
(c) Niblack (d) Otsu

Fig. C.5.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [1, 6.0974, 0.5332, 0.0244, 0.0011].

(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

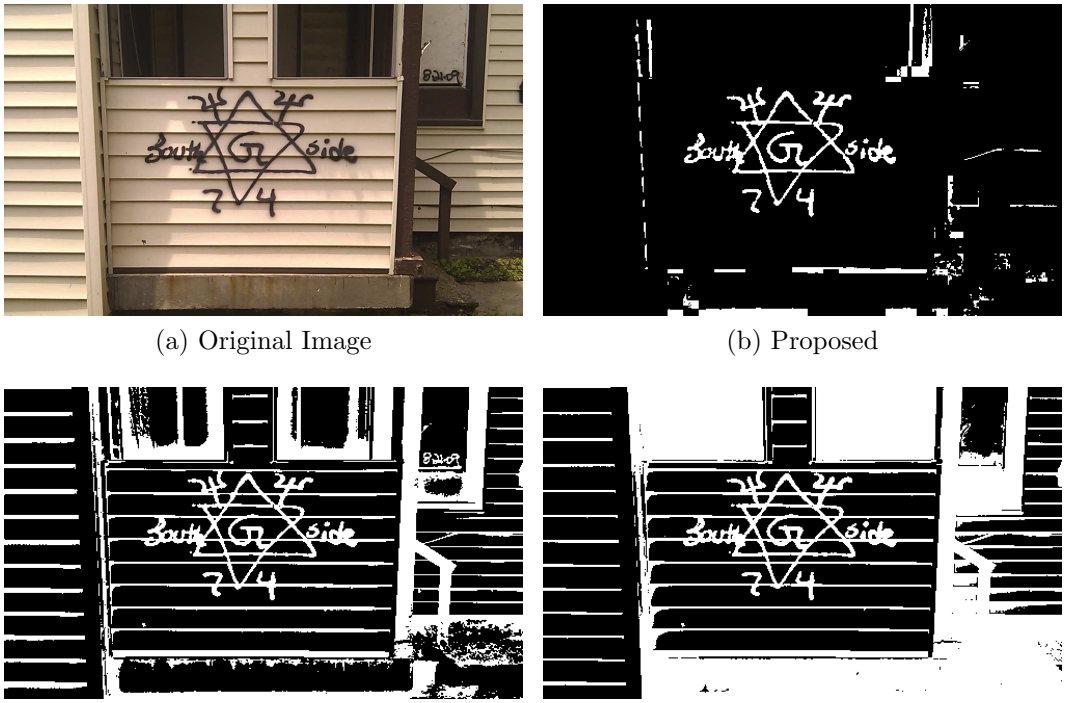
Fig. C.6.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [0, 6.1730, 0.7483, 0.0093, 0.0037].



(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

Fig. C.7.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [1, 0.1145, 0.2670, 0.0080, 0.0028].



(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

Fig. C.8.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [0, 0.1848, 0.2120, 0.0656, 0.0017].

(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

Fig. C.9.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [0, 4.8869, 0.1329, 1.2905, 0.0029].

(a) Original Image

(b) Proposed

(c) Niblack

(d) Otsu

Fig. C.10.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [0, 3.6070, 0.1894, 2.3252, 0.0013].

(a) Original Image

(b) Proposed

(c) Niblack

(d) Otsu

Fig. C.11.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [0, 2.7925, 0.3618, 0.1469, 0.0028].

(a) Original Image

(b) Proposed

(c) Niblack

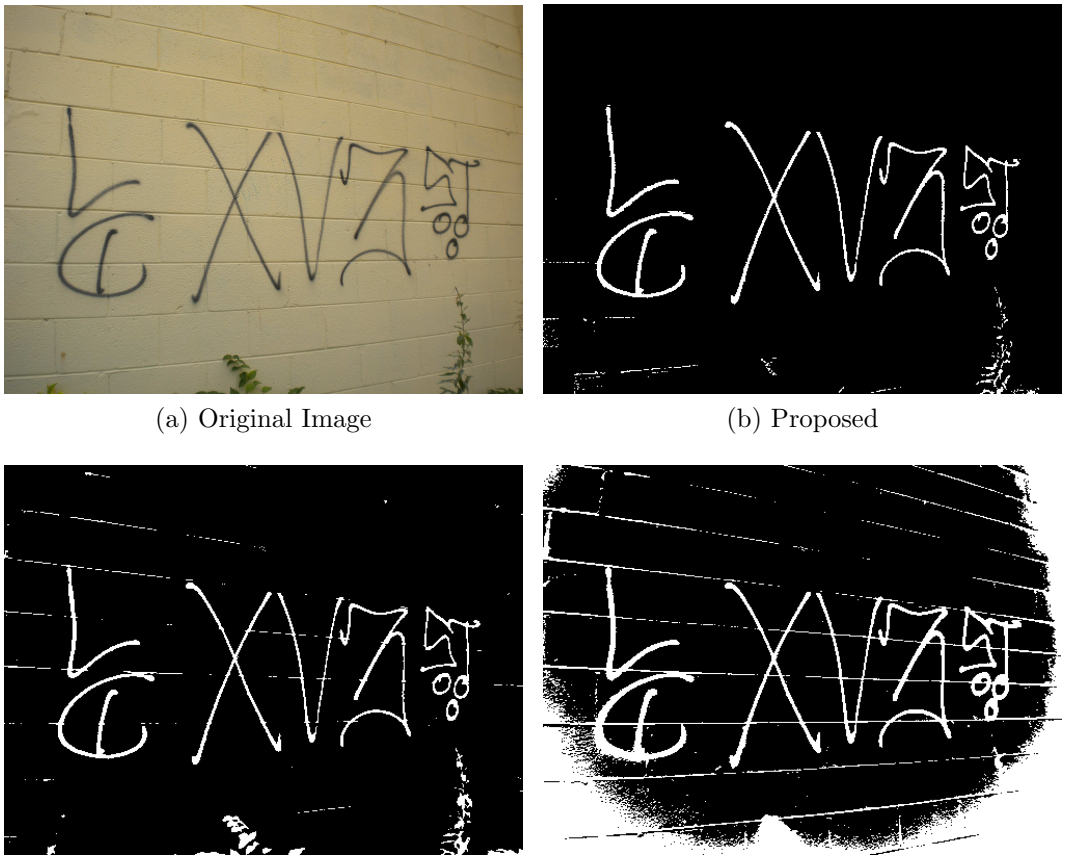
(d) Otsu

Fig. C.12.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [0, 1.0472, 0.2784, 2.6779, 0.0161].

(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

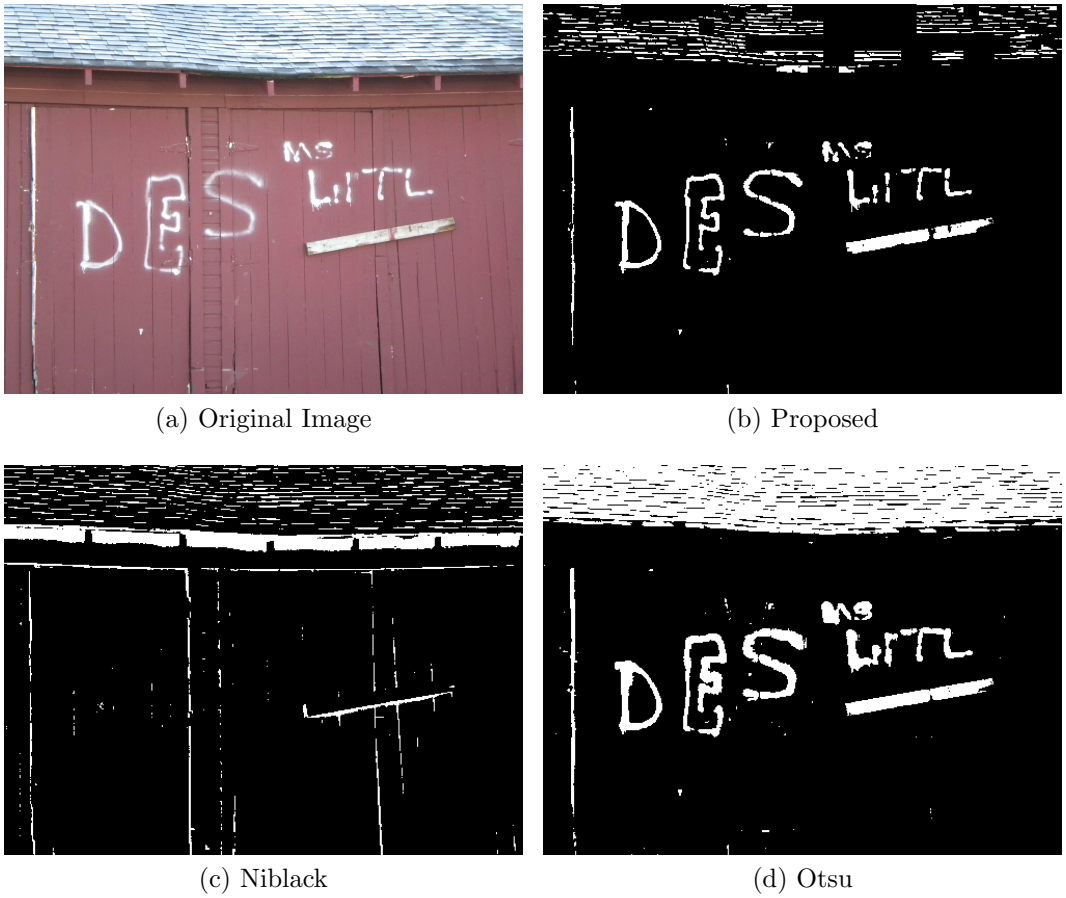
Fig. C.13.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [1, 3.5358, 0.4344, 0.0016, 0.0028].



(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

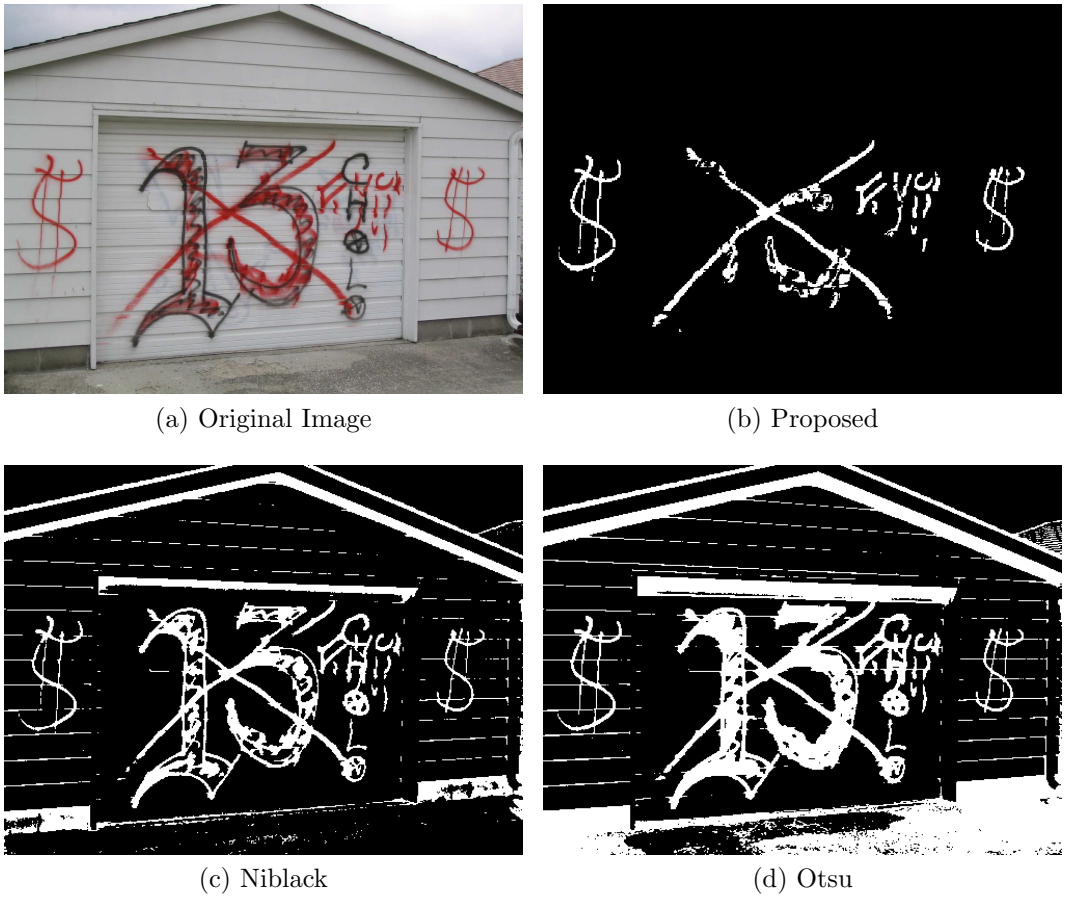
Fig. C.14.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [0, 0.7854, 0.3680, 0.0250, 0.0019].



(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

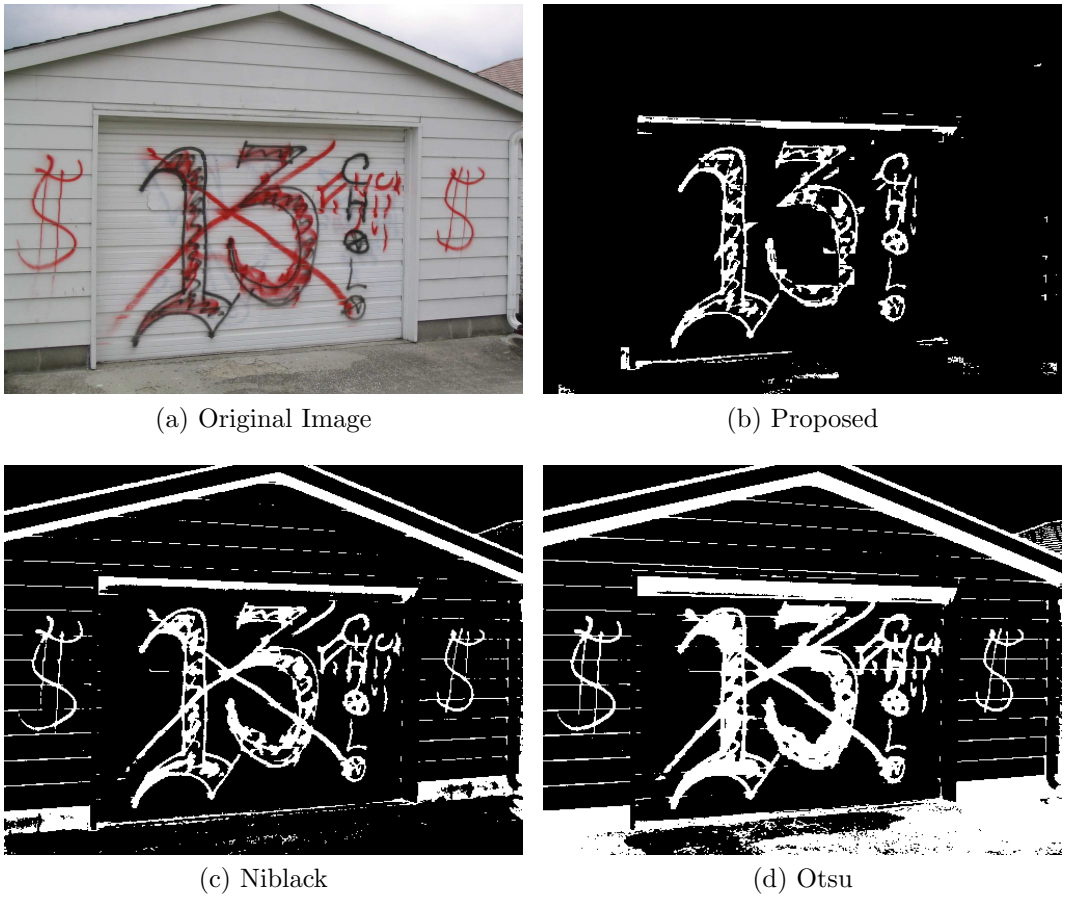
Fig. C.15.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [0, 4.8171, 0.8821, 0.3069, 0.0046].



(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

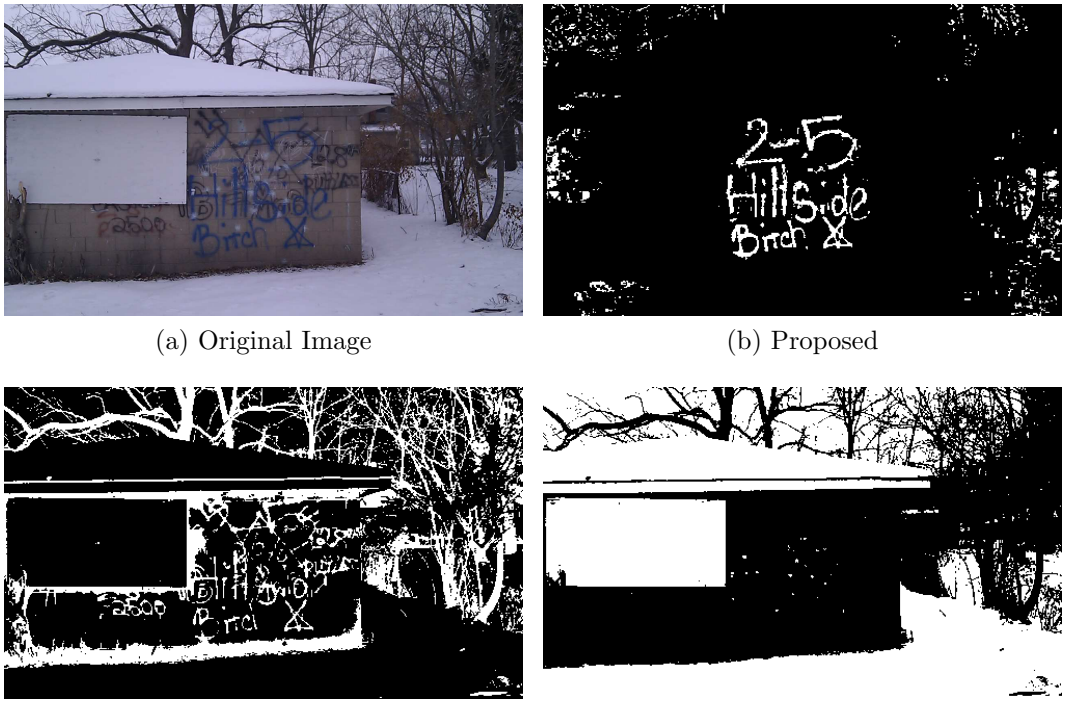
Fig. C.16.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [1, 0.0423, 0.3018, 0.0012, 0.0018].



(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

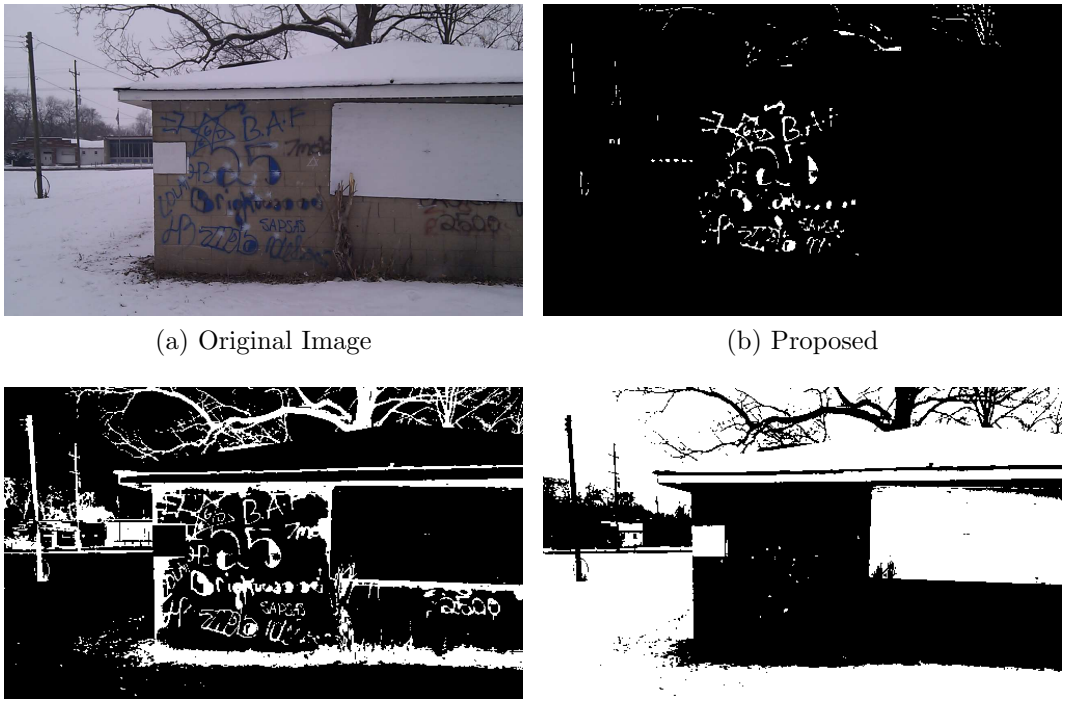
Fig. C.17.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [0, 0.1309, 0.2317, 0.3181, 0.0093].



(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

Fig. C.18.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [1, 4.0075, 0.1993, 0.0021, 0.0015].



(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

Fig. C.19.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [1, 3.9924, 0.1886, 0.1030, 0.0014].

(a) Original Image (b) Proposed

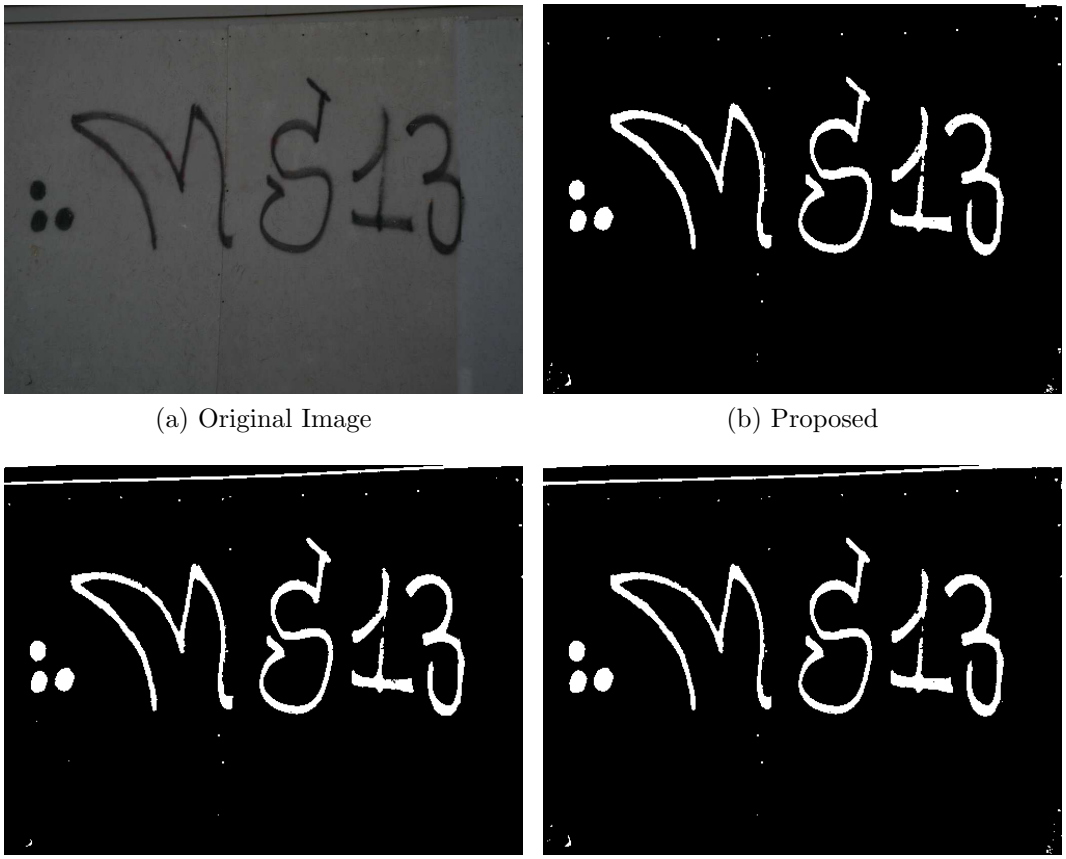
(c) Niblack (d) Otsu

Fig. C.20.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [1, 0.1496, 0.3147, 0.0049, 0.0022].

(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

Fig. C.21.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [0, 1.0472, 0.1529, 1.7701, 0.0005].



(a) Original Image (b) Proposed

(c) Niblack (d) Otsu

Fig. C.22.: For Proposed Method: [boolHL, medH, medY, varH, varY] = [0, 2.6180, 0.1305, 2.3481, 0.0019].

D. GARI DATABASE TABLES

This Appendix describes the GARI database tables in more detail.

Table D.1: EXIF data fields in Table *images*.

EXIF field	**Description**
filesize	Size of the image (bytes)
filedatetime	Date and time of capture
resolutionheight	Height of image (px)
resolutionwidth	Width of image (px)
focallength	Focal Length of camera's optical system
isoequiv	ISO equivalent value used
cameramake	Camera make
cameramodel	Camera model
gpsaltitude	GPS altitude
gpslongitude	GPS longitude
gpslatitude	GPS latitude
xresolution	DPI in the width direction
yresolution	DPI in the height direction
ycbcrpositioning	Position of the YCbCr components
fnumber	F number
compressedbitsperpixel	Compressed bits per pixel
exposuretime	Exposure time (seconds)
exposurebias	Exposure bias (APEX)
aperture	Lens aperture (APEX)
meteringmode	Metering mode

flash	Status of flash when the image was shot
interoperabilityoffset	Interoperability offset
sensingmethod	Sensing method
customrendered	Use of special processing on image data
whitebalance	White balance
digitalzoomratio	Digital zoom ratio
exposuremode	Exposure mode

Table D.2: Image location fields in Table *images*.

Field	**Description**
country	Country (given GPS coordinates)
state	State (given GPS coordinates)
county	County (given GPS coordinates)
city	City (given GPS coordinates)
zip	ZIP code (given GPS coordinates)
address	Address (given GPS coordinates)

Table D.3: Graffiti analysis fields in Table *images*.

Field	**Description**
gangnameia	Gang name from IA [1]
gangnamegt	Gang name from GT [2]
gangidia	Gang ID from IA
gangidgt	Gang ID from GT
gangmembernameia	Gang member name from IA
gangmembernamegt	Gang member name from GT

[1]IA: Image Analysis
[2]GT: Ground Truth

gangmemberidia	Gang member ID from IA
gangmemberidgt	Gang member ID from GT

Table D.4: Image information fields in Table *images*.

Field	**Description**
imageid	Image ID
path	Path to the image file
firstrespondername	First responder name
firstresponderid	First responder ID
comment	Comments about graffiti
webupload	File uploaded from desktop version (boolean)
realcoords	Image has real GPS coordinates (boolean)
filedatetimeupload	Date and time the file was uploaded to the database
lastmodified	Date and time a fields was last modified
lastmodifiedname	First responder that last modified a field
istattoo	Boolean to indicate if image is graffiti or tattoo
isprison	Boolean to indicate if image was taken at a prison
prisonname	Name of the prison where the image was taken

Table D.5: User information fields in Table *users*.

Field	**Description**
id	User ID
password	MD5 hash of user's password
name	User's name
admin	User is administration (boolean)
first	First login (boolean)

gmail	Gmail address
email	Alternative email address
affiliation	User affiliation
android	Has Android application (boolean)
comments	Comments about user

Table D.6: Image blobs information fields in Table *imageBlobs*.

Field	**Description**
imageid	Image ID
blobid	Blob ID for a particular image ID
componentid	Component ID for a particular blob ID
colorid	Color ID for a particular component ID
crossedout	Boolean to determine if the component is crossed-out
upsidedown	Boolean to determine if the component is upside-out

E. MERGE DATABASE TABLES

This Appendix describes the MERGE database tables in more detail.

Table E.1: EXIF data fields in Table *images*.

EXIF field	**Description**
filesize	Size of the image (bytes)
filedatetime	Date and time of capture
resolutionheight	Height of image (px)
resolutionwidth	Width of image (px)
focallength	Focal Length of camera's optical system
isoequiv	ISO equivalent value used
cameramake	Camera make
cameramodel	Camera model
gpsaltitude	GPS altitude
gpslongitude	GPS longitude
gpslatitude	GPS latitude
xresolution	DPI in the width direction
yresolution	DPI in the height direction
ycbcrpositioning	Position of the YCbCr components
fnumber	F number
compressedbitsperpixel	Compressed bits per pixel
exposuretime	Exposure time (seconds)
exposurebias	Exposure bias (APEX)
aperture	Lens aperture (APEX)
meteringmode	Metering mode

flash	Status of flash when the image was shot
interoperabilityoffset	Interoperability offset
sensingmethod	Sensing method
customrendered	Use of special processing on image data
whitebalance	White balance
digitalzoomratio	Digital zoom ratio
exposuremode	Exposure mode

Table E.2: Image location fields in Table *images*.

Field	**Description**
country	Country (given GPS coordinates)
state	State (given GPS coordinates)
county	County (given GPS coordinates)
city	City (given GPS coordinates)
zip	ZIP code (given GPS coordinates)
address	Address (given GPS coordinates)

Table E.3: Image information fields in Table *images*.

Field	**Description**
imageid	Image ID
path	Path to the image file
firstrespondername	First responder name
filedatetimeupload	Date and time the file was uploaded to the database
issign	Boolean to indicate if image is sign or scene

Table E.4: User information fields in Table *users*.

Field	**Description**
id	User ID
password	MD5 hash of user's password
name	User's name
admin	User is administration (boolean)
first	First login (boolean)
gmail	Gmail address
email	Alternative email address
affiliation	User affiliation
android	Has Android application (boolean)
comments	Comments about user

Table E.5: Fields in Table *class*.

Field	**Description**
clid	Class ID
text	Text describing class number and name
clnumber	Class number

Table E.6: Fields in Table *colorids*.

Field	**Description**
colorid	Color ID
colorname	Color name

Table E.7: Fields in Table *colorpages*.

Field	Description
colorid	Color ID
guide	Guide page number

Table E.8: Fields in Table *placard*.

Field	Description
pid	Placard ID
unid	UNID
clid	Class ID
sid	Symbol ID

Table E.9: Fields in Table *symbol*.

Field	Description
sid	Symbol ID
text	Symbol description

Table E.10: Fields in Table *textcolors*.

Field	Description
textid	Text ID for hazardous material types
colorid	Color ID

Table E.11: Fields in Table *textids*.

Field	Description
textid	Text ID
text	Hazardous material description

Table E.12: Fields in Table *textpages*.

Field	**Description**
textid	Text ID
guide	Guide page number

Table E.13: Fields in Table *unids*.

Field	**Description**
unids	UNID
guide	Guide page number
material	Material type
iso	Included in the International Organization for Standardization (ISO) (boolean)

Table E.14: Fields in Table *vw01_orange_page*.

Field	**Description**
guide_number_cd	Guide page number
guide_page_name_txt	Guide page title
category_txt	Hazmat sign category
sub_category_txt	Hazmat sign subcategory
detail_txt	Page details

Table E.15: Fields in Table *vw03_yellow_page*.

Field	**Description**
un_number	UNID
guide_number_cd	Guide page number
polymerization_ind	Polymerization index

dangerous_good_name_txt	Dangerous good description
dangerous_good_id	Dangerous good ID

Table E.16: Fields in Table *vw05_water_reactive_materials*.

Field	**Description**
un_number	UNID
guide_number_cd	Guide page number
dangerous_good_name_txt	Dangerous good description
chemical_symbol	Chemical symbol
tih_gas_produced	Toxic-by-Inhalation (TIH) gas produced
dangerous_good_id	Dangerous good ID
polymerization_ind	Polymerization index

Table E.17: Fields in Table *vw06_tiiapad*.

Field	**Description**
dangerous_good_id	Dangerous good ID
dangerous_good_name_txt	Dangerous good description
un_number	UNID
circumstance_type_txt	Situation when condition applies
guide_number_cd	Guide page number
polymerization_ind	Polymerization index
simetric	Small spills - Isolation distance (metric)
spdmetric	Small spills - Protective distance - Day (metric)
spnmetric	Small spills - Protective distance - Night (metric)
limetric	Large spills - Isolation distance (metric)

lpdmetric	Large spills - Protective distance - Day (metric)
lpnmetric	Large spills - Protective distance - Night (metric)
siimperial	Small spills - Isolation distance (imperial)
spdimperial	Small spills - Protective distance - Day (imperial)
spnimperial	Small spills - Protective distance - Night (imperial)
liimperial	Large spills - Isolation distance (imperial)
lpdimperial	Large spills - Protective distance - Day (imperial)
lpnimperial	Large spills - Protective distance - Night (imperial)

F. GARI IMAGE ACQUISITION PROTOCOL

This Appendix describes the protocol used for acquiring test images for the GARI database. The images are used for testing various functions of the GARI system.

- Persons involved
  - 2 GARI staff members
  - 1 or more persons from Police Department
- Equipment/Materials needed
  - Pens or pencils
  - 2 Digital Camera (1MPx and above)
  - 2 Tripods
  - 2 Mobile Telephone with Android OS
    - Built-in camera (1MPx and above)
    - GPS receiver
    - optional: Data plan
  - 1 GPS receiver
  - Graffiti Information Forms
  - Fiducial Markers
  - Image Checklist
  - 1 Purdue University owned laptop
  - 1 External hard drive

1) Preliminaries (Internet connection required)
   a) Check time setting on the two Android mobile telephones, the two digital cameras, and the GPS receiver using the Purdue University owned laptop, and ensure they are in sync with the GARI server.

b) Make sure the two Android mobile telephones, the two digital cameras, and the GPS receiver batteries are fully charged.

c) Verify all equipment/materials above are available.

d) Make sure the settings of the two digital cameras are set to default by finding the appropriate menu option.

e) Turn flash feature off on the two Android mobile telephones built-in cameras and the two digital cameras.

f) Make sure zoom and macro features are not enabled on the two Android mobile telephones built-in cameras and the two digital cameras.

g) Assign each person an ID number, and record it on the Graffiti Information Form.

h) Record person's name and affiliation on the Graffiti Information Form.

2) Set up environment

a) Stand up in front of the graffiti, far enough so that the cameras can capture all the content, preferably perpendicular to the surface containing the graffiti. Some angle margin is permitted (θ spherical degrees), as shown in Figure F.1 and Figure F.1. This angle show be small enough so that the graffiti contents can be identified properly.

b) Make sure weather condition does not prevent seeing the graffiti.

c) Place the fiducial marker in a spot that would be 20 inches away and parallel to the surface containing the graffiti, as shown in Figure F.1 and Figure F.2. It should not block the graffiti contents.

d) Make sure there are not any objects between the camera and the graffiti that obstruct partially or totally the view of the graffiti.

e) Record Date (MM/DD/YYYY), Time (HH:MM:SS) and GPS coordinates (latitude, longitude and altitude, with six digit precision) on the Graffiti Information Form. Obtain the information from the GPS receiver.

f) Record neighborhood description on the Graffiti Information Form. Specify street name(s) and landmarks in the area near the graffiti.

g) Proceed to take image. For each graffiti, take six images, using

- Android mobile telephone 1
- Android mobile telephone 1
- Android mobile telephone 2
- Digital camera 1 with tripod
- Digital camera 1 without tripod
- Digital camera 2 with tripod
- Digital camera 2 without tripod

h) For each graffiti, record the device(s) used on the Graffiti Information Form.

3) Taking an image of a graffiti

3.1) Taking image of a graffiti using an Android mobile telephone

a) Launch GARI application on the Android mobile telephone and assign an Image Taker ID, corresponding to the one assigned in step 1. Preliminaries.

b) Select the "Capture Image" option from the GARI application main menu. The camera activity is then initialized.

c) Prepare for taking the image (position of the camera as desired, within the recommended distance and angle from the graffiti). Make sure all the contents of the graffiti and the entire fiducial marker can be seen on the device screen.

d) Take an image of the graffiti, trying to maintain the device's position, as much as possible.

e) If the image does not meet the requirements noted in the Image Checklist, the image should be retaken.

f) If location available through WiFi/GSM/GPS the GPS coordinates will be automatically stored in the image. If no location method available,

will receive a message: "No NETWORK/GPS found. Check coordinates manually!". Ignore it, since the GPS coordinates have already been recorded on the Graffiti Information Form.

g) Crop the image if desired.

h) Select the "Send to Server" option from the GARI application main menu. If no Internet connection available, will receive a message: "No internet connection available". It means the image has not been uploaded to the server. However, the image is still in the Android mobile telephone SD card, and it can be copied to a computer at the end of the session (Section 5.a of the protocol), and uploaded in the future. If the image has not been uploaded to the server, check the box "Not Successfully Uploaded" on the Graffiti Information Form.

3.2) Taking image of a graffiti using a digital camera

a) If a tripod is used, attached it to the digital camera, and adjust it so the digital camera is at the same position as if it is held without using the tripod.

b) Prepare for taking the image (position of the camera as desired, within the recommended distance and angle from the graffiti). Make sure all the contents of the graffiti and the entire fiducial marker can be seen on the device screen.

c) Take an image of the graffiti, trying to maintain the device's position, as much as possible.

4) Completing the Graffiti Information Form (Figure F.3)

a) Fill the "Ground-truth graffiti information" section on the Graffiti Information Form with ground-truth information associated with the graffiti, if known. It includes:

- Graffiti color(s): color or colors of the graffiti contents.

- Gang Name(s): name of the gang or gangs that participated on the drawing of the graffiti.
- Gang Member(s): name of the gang member or gang members that participated on the drawing of the graffiti.
- Target Gang Name(s): name of the gang or gangs that are targeted in the graffiti.
- Target Gang Member(s): name of the gang member or gang members that are targeted in the graffiti.
- Symbol(s): description of the symbol(s) in the graffiti, including color, position in the graffiti (e.g. next to the gang name), orientation (e.g. upside down fork), and possible meaning.
- Other content(s): description of other relevant contents of the graffiti (e.g. crossed letters, nicknames), including color, position in the graffiti (e.g. crossed C on the right of BERO), and possible meaning.
- Comments: additional information of the graffiti that does not fit in the previous subsections of the "Ground-truth graffiti information" section.

b) Fill the "General Comments" section on the *Graffiti Information Form* with additional comments that do not fit in all the previous sections.

5) End of the session procedures

a) Copy all the images taken with the Android mobile telephones (stored in the GARI folder) and with the two digital cameras to a Purdue University owned laptop and to an external hard drive.

b) Take cards out of the digital cameras and reformat them.

c) Ensure the Purdue University owned laptop and the two digital cameras are synced.

d) Recharge laptop and camera batteries.

e) Store fiducial markers and other materials in a safe place for later use.

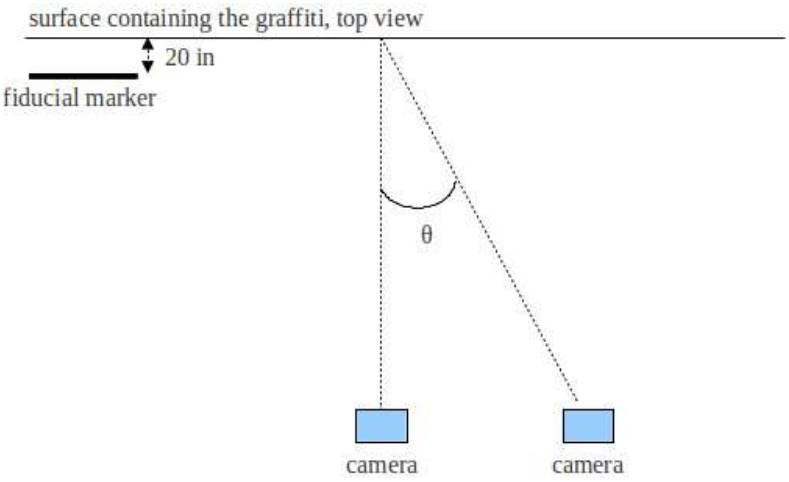


Fig. F.1.: Top view of the setup environment.

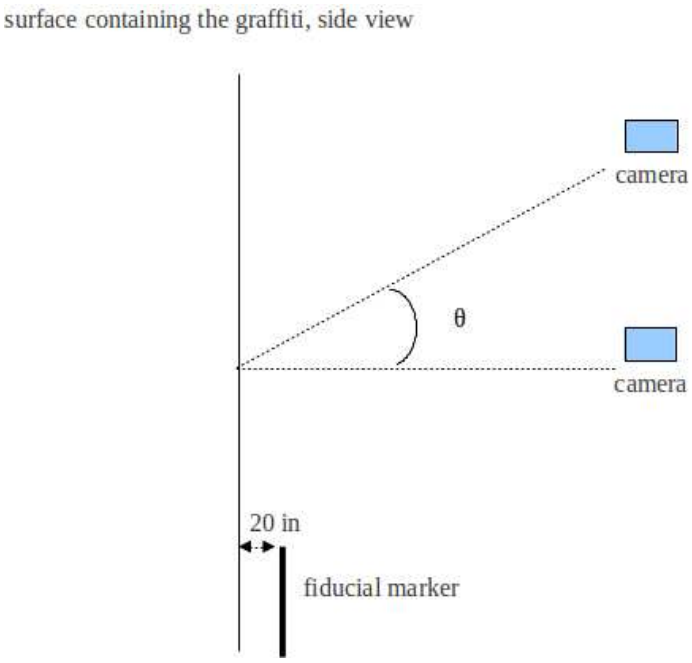


Fig. F.2.: Side view of the setup environment.

Graffiti Information Form

First Responder information

Name: ID: Affiliation:

Basic image information

Date: / / Time: : :

GPS Coordinates

Latitude: Longitude: Altitude:

Neighborhood Description:

Device(s) used

☐ Android Mobile Telephone ☐ Camera with tripod ☐ Camera without tripod

---

Ground-truth graffiti information

Color(s):

Gang Name(s):

Gang Member(s):

Target Gang Name(s):

Target Gang Member(s):

Symbol(s):

Other Content(s):

Comments:

General Comments:

Fig. F.3.: Graffiti Information Form.

G. MERGE IMAGE ACQUISITION PROTOCOL

This Appendix describes the protocol used for acquiring test images for the MERGE database. The images are used for testing various functions of the MERGE system.

- Persons involved
  - 1 MERGE staff member
- Equipment/Materials needed
  - Pens or pencils
  - 1 Mobile Telephone with Android OS
    - Built-in camera (1MPx and above)
    - 3G/4G/WiFi data connection
    - GPS
  - 1 Digital Camera with Android OS
    - 3G/4G/WiFi data connection
    - GPS
  - Image Recording Forms
  - External Hard Drive

1) Preliminaries (Internet connection required)
   a) Check Date and Time settings on the Android mobile telephone and the digital camera, and ensure date, time, and time zone are set to automatic (network-provided).
   b) Make sure the Android mobile telephone and the digital camera's batteries are fully charged.
   c) Make sure the GPS is enabled on the Android mobile telephone and the digital camera.

d) Verify all equipments/materials above are available.

e) Turn flash feature off on the Android mobile telephone and the digital camera.

f) Note: The Image Taker will need to fill out an Image Recording Form for each hazmat sign.

2) Set up environment

a) Stand in front of the hazmat sign, far enough so that the camera can capture all the content, up to 200 feet from the sign for the Android mobile phone, and up to 500 feet from the sign for the digital camera. Stand preferably perpendicular to the surface containing the sign. Limited angles are permitted (45 degrees), as shown in Figure G.1.

b) Make sure weather conditions do not obstruct the view of the hazmat sign.

c) Make sure there are no objects between the camera and the hazmat sign that partially or completely obstruct the view of the hazmat sign.

3) Taking Images of Hazmat Signs

a) Launch the MERGE application on the Android mobile telephone and the digital camera, and login using the Image Taker's ID and password. If this is the first time that the Image Taker is logging into the application, an Internet connection will be required to connect with the MERGE database on the server. From then on, the Image Taker's credential will be stored on the Android device for future use without an Internet connection.

b) Select the "Capture Image" option from the MERGE main screen. The camera activity is then initialized. Note that a new directory with the name MERGE will be created on the Android device's image gallery, where all the images taken using the MERGE application will be stored. Please refer to this directory when copying the images to the external hard drive (Section 5a).

c) Prepare for taking the image (position the camera as desired, within the recommended distance and angle from the hazmat sign). Make sure all the contents of the hazmat sign can be seen on the device screen.

d) Take an image of the hazmat sign, trying to hold the device as much as stable. The image can be retaken as many times as needed by tapping on the retake option on the camera activity.

e) Tap on the OK button on the camera activity to save the current image. The image will be automatically uploaded to the server and analyzed. The Image Taker should see a notification dialog with the text "Uploading image..." followed by another notification dialog with the text "Analyzing image...". If no Internet connection is available at the time, a warning dialog with the text "No Internet connection available" will be shown to the Image Taker. However, the image is stored in the Android device, and it can be uploaded and analyzed in the future using the "Browse Image" option from the MERGE main screen. If the image has not been uploaded to the server, check the box "Not Successfully Uploaded" on the Image Recording Form.

f) If no Internet connection is available at the time, a warning dialog with the text "No Internet connection available" will be shown to the Image Taker. In this case, the captured image is stored in the device, and it can be uploaded and analyzed in the future using the "Browse Image" option from the MERGE main screen.

g) Please take different images for the same sign, at different distances (10-150 ft) and angles of view (0-45°), and then write down an Image ID shown on the top bar / pop-up window on the result screen, an approximate Angle of View between your viewpoint and the perpendicular plane of the hazmat sign's surface, and an approximate Distance from your viewpoint to the hazmat sign on the Image Recording Form (e.g., 123456, 15°, and 125 ft).

h) Please take at least one image with No Zoom when using the digital camera, and then check the box "No Zoom" on the Image Recording Form. Also

take some images using the Optical Zoom when using the digital camera (NO Digital Zoom), and then check the box "Zoom" and mark on an approximate Zoom Value in a box on the Image Recording Form (e.g., 3/4 of the entire optical zoom range).

4) Completing the Image Recording Form (Figure G.2)

   a) Record Date (MM/DD/YYYY), Starting Time (HH:MM:SS), the Make and Model of the device used to capture the images (e.g., HTC Desire) and the Image Taker's Name and Affiliation on the Image Recording Form.

   b) Complete the "Ground Truth Information" section on the Image Recording Form with ground-truth information associated with each hazmat sign in the captured image. This includes:

      - The Total number of existing hazmat signs in the captured image
      - For each existing hazmat sign
        - Hazmat sign number of an existing hazmat sign in the captured image
        - Color(s): color(s) found in the hazmat sign (NOT including hazmat sign frame)
        - UN Identification number (UNID) (Figure G.3a)
        - Symbol (Figure G.3b)
        - Class (Figure G.3c)
        - Text (Figure G.3d)
        - Comments: Additional information of the hazmat sign that does not fit in the previous fields.

   c) Complete the "Image Analysis Results" section on the Image Recording Form with information retrieved from the server after a captured or browsed image has been analyzed. This includes:

      - The Image ID of the captured image
      - The Total number of highlighted hazmat signs from image analysis
      - For each returned hazmat sign

- Hazmat sign number of a highlighted hazmat sign shown in the result screen
- Color(s): color(s) shown in the result screen
- Text: text shown in the result screen
- No hazmat signs found: Check this box if a dialog containing "No hazmat signs found" is shown to the Image Taker after uploading an image to the server, meaning that no hazmat signs have been found in the current image.

Figures G.4 and G.5 show two examples of completed Image Recording Forms for two different cases shown in Figure G.6.

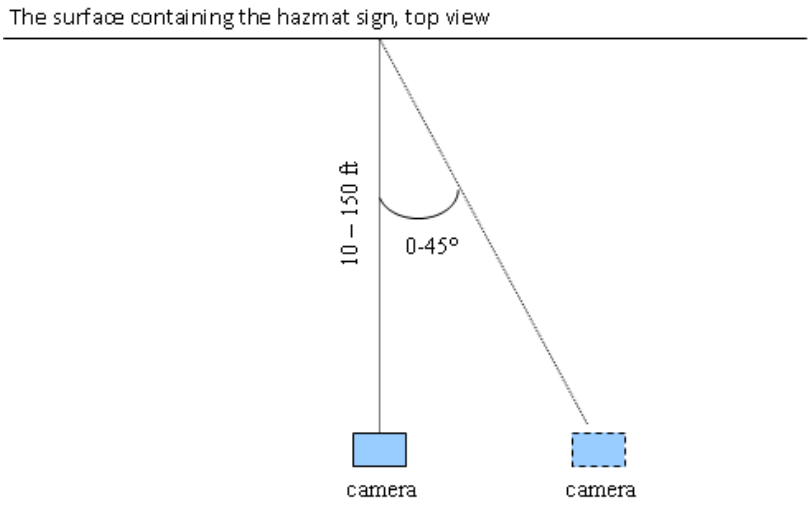


Fig. G.1.: Top view of the setup environment.

Image Recording Form

Image Taker Name: **ID:** **Affiliation:**

Date: / / **Starting Time:** : :

Device Make: **Device Model:**

Ground Truth Information			Angle of View		°	Distance		ft
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments
Image Analysis Results			No Zoom [ ]		Zoom [ ]		1/4 1/2 3/4 Full	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found
								[ ]

Ground Truth Information			Angle of View		°	Distance		ft
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments
Image Analysis Results			No Zoom [ ]		Zoom [ ]		1/4 1/2 3/4 Full	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found
								[ ]

Ground Truth Information			Angle of View		°	Distance		ft
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments
Image Analysis Results			No Zoom [ ]		Zoom [ ]		1/4 1/2 3/4 Full	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found
								[ ]

Ground Truth Information			Angle of View		°	Distance		ft
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments
Image Analysis Results			No Zoom [ ]		Zoom [ ]		1/4 1/2 3/4 Full	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found
								[ ]

Fig. G.2.: Image Recording Form.

(a) UNID (b) Symbol (c) Class (d) Text

Fig. G.3.: Hazmat sign identifiers.

Ground Truth Information			Angle of View		5 °	Distance		80 ft	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments	
120130	1	2	WHITE	1017	Skull	2	No Text		
Image Analysis Results			No Zoom [X]		Zoom [ ]		1/4 1/2	3/4	Full
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found	
120130	1	2	WHITE	N/A	N/A	N/A	No Text	[ ]	

Ground Truth Information			Angle of View		5 °	Distance		80 ft	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments	
120130	2	2	WHITE	1017	Skull	2	No Text		
Image Analysis Results			No Zoom [X]		Zoom [ ]		1/4 1/2	3/4	Full
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found	
120130	2	2	WHITE	N/A	N/A	N/A	No Text	[ ]	

Fig. G.4.: Example of Completed Image Recording Form for Figure G.6 (left).

Ground Truth Information			Angle of View		15 °	Distance		125 ft
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	Comments
171215	1	1	RED	1267	Flame	3	No Text	
Image Analysis Results			No Zoom [ ]		Zoom [X]		1/4 1/2 3/4 Full	
Image ID	Hazmat Sign Number	Total Num. of Hazmat Signs	Color(s)	UNID	Symbol	Class	Text	No hazmat signs found
171215	1	1						[X]

Fig. G.5.: Example of Completed Image Recording Form for Figure G.6 (right).

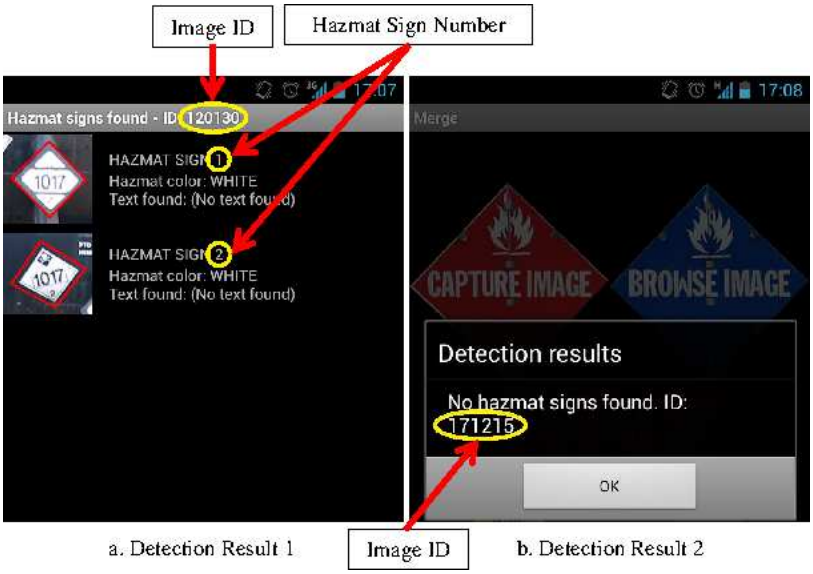


Fig. G.6.: Screenshots for hazmat sign found (left) and not found (right).

VITA

VITA

Albert Parra Pozo was born in Barcelona, Catalonia, Spain. He received the B.S. degree in Superior Telecommunications Engineering from the Universitat Politècnica de Catalunya (UPC) in 2010. He was a visitor scholar in the Video and Image Processing Laboratory (VIPER) at Purdue University between 2009 and 2010. He received the M.S. degree in Electrical and Computer Engineering from Purdue University in 2011. He joined the Ph.D program in Electrical and Computer Engineering at Purdue University in January 2012. He has worked as a Research Assistant in the VIPER lab under the direction of Professor Edward J. Delp since 2010, being sponsored by the U.S. Department of Homeland Security's VACCINE Center. He is a student member of the IEEE and the IEEE Signal Processing Society, and student member of the Association for Computing Machinery (ACM). He has been reviewer of the IEEE Journal on Transactions on Multimedia.

Albert Parra Pozo's publications are:

1. Chang Xu, Ye He, **Albert Parra**, Nitin Khanna, Carol J. Boushey, and Edward J. Delp, "Image-Based Food Volume Estimation," *Proceedings of the ACM International Conference on Multimedia*, October 2013, Barcelona, Spain.
2. Bin Zhao, **Albert Parra** and Edward J. Delp, "Mobile-Based Hazmat Sign Detection System," *Proceedings of the IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 735-738, December 2013, Austin, TX.
3. **Albert Parra**, Bin Zhao, Joonsoo Kim and Edward J. Delp, "Recognition, Segmentation and Retrieval of Gang Graffiti Images on a Mobile Device," *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, pp. 178-183, November 2013, Waltham, MA.
4. **Albert Parra**, Bin Zhao, Andrew Haddad, Mireille Boutin and Edward J. Delp, "Hazardous Material Sign Detection and Recognition," *Proceedings of the IEEE International Conference on Image Processing*, pp. 2640-2644, September 2013, Melbourne, Australia.
5. Mark Q. Shaw, **Albert Parra**, Jan P. Allebach, "Improved Video Compression using Perceptual Modeling," *Proceedings of the IS&T Color and Imaging Conference*, pp. 9-14, November 2012, Los Angeles, CA.
6. Mark Q. Shaw, **Albert Parra**, Jan P. Allebach. Techniques for Video Compression. U.S. Patent Application. PCT/US12/48514, filed July 2012. Patent Pending.
7. **Albert Parra**, Mireille Boutin and Edward J. Delp, "Location-Aware Gang Graffiti Acquisition and Browsing on a Mobile Device," *Proceedings of the IS&T/SPIE Electronic Imaging on Multimedia on Mobile Devices*, pp. 830402-1-13, January 2012, San Francisco, CA.
8. **Albert Parra**, Andrew W. Haddad, Mireille Boutin, Edward J. Delp, "A Hand-Held Multimedia Translation and Interpretation System for Diet Management,"

Proceedings of the IEEE International Workshop on Multimedia Services and Technologies for E-health in conjunction with the International Conference on Multimedia and Expo (ICME), pp. 1-6, July 2011, Barcelona, Spain.

9. **Albert Parra**, Andrew W. Haddad, Mireille Boutin, Edward J. Delp, "A Method for Translating Printed Documents Using a Hand-Held Device," *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1-6, July 2011, Barcelona, Spain.

10. **Albert Parra**, Mireille Boutin, Edward J. Delp, "iPod-Based System for the Automatic Translation and Interpretation of Spanish Language Menus," *Demonstration in Light-Weight Image Processing on Cellular Phones and PDAs, Show and Tell Demonstrations of the 35th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, March 2010, Dallas, TX.

Visual Saliency Models Based on Spectrum Processing

Bin Zhao and Edward J. Delp
Video and Image Processing Laboratory (VIPER)
School of Electrical and Computer Engineering
Purdue University
West Lafayette, Indiana, USA
redpill.ecn.purdue.edu/∼zhao61

Abstract

Some visual saliency models have been proposed to describe how the human visual system perceives and processes visual information. In this paper we describe four frequency domain visual saliency models based on new spectrum processing methods. The four saliency models are the Gamma Corrected Spectrum (GCS) model, the Gamma Corrected Log Spectrum (GCLS) model, the Gaussian Filtered Spectrum (GFS) model, and the Gaussian Filtered Log Spectrum (GFLS) model. A set of saliency map candidates are generated by inverse transform of a set of modified spectrums. An output saliency map is selected by minimizing the entropy among the set of saliency map candidates. Extension of these models are also decribed using various color spaces. Experimental results show that four extensions of our GCS, GCLS, GFS, and GFLS models are more accurate and efficient than some state-of-the-art saliency models in predicting eye fixation on standard image datasets.

1. Introduction

Visual saliency has been modeled as a preprocessing step of the human visual system that selects the important visual information in a scene [21]. Visual saliency is the perceptual process that makes attractive objects "stand out" from their surroundings in the low-level human visual system. A master map of the "salient objects" [30] or a saliency map [21] is generated by the low-level vision system to indicate the locations of salient regions in a scene. High-level, cognitive and more complex operations are mostly focused on the selected salient regions. Visual saliency has been investigated in multiple fields including cognitive psychology, neuroscience, computer vision, and image and video processing. Visual saliency models are used in many applications including image and video compression [7, 19], content-aware image resizing [2], object extraction [15], object recognition [31], and traffic sign image analysis [23].

Some visual saliency models have been proposed to describe how the human visual system perceives and processes visual information [12, 29, 4, 5]. The Saliency-Based Visual Attention (SBVA) model is proposed in [20] using intensity, color and orientation features with a subsampled Gaussian pyramid. In [16] a Graph-Based Visual Saliency (GBVS) method forms an activation map from each feature map based on graph theory. The Attention based on Information Maximization (AIM) model is presented in [6] using Independent Component Analysis (ICA) based feature extraction, joint likelihood, and self-information. A Frequency-Tuned Saliency Detection (FTSD) model is introduced by [1] using low-level features of color and luminance. A Spectral Residual (SR) approach is proposed in [18] using the spectral residual of the log spectrum of an image to generate its saliency map. Two similar saliency models are developed using the Phase spectrum of the Fourier Transform (PFT) [13] and the Phase spectrum of Quaternion Fourier Transform (PQFT) [14] to predict salient regions in the spatio-temporal domain. Two biologically plausible visual saliency approaches, Frequency domain Divisive Normalization (FDN) and Piecewise FDN (PFDN) methods, are proposed in [3], where PFDN shows better performance and provides better biological plausibility. In [17] an Discrete Cosine Transform (DCT) based Image Signature (IS) method generates a saliency map using the inverse DCT of the signs of the cosine spectrum for image figure-ground separation. A Quaternion DCT (QDCT) based image signature approach is developed by [25] using the signum function and the inverse QDCT to compute a visual saliency map. A saliency detector based on the Scale-Space Analysis (SSA) is presented in [22] using the convolution of the amplitude spectrum of the Hypercomplex Fourier Transform (HFT) with a set of Gaussian kernels.

The focus of this paper is to investigate low-complexity bottom-up visual saliency models using spectral analysis approaches. We generalize existing visual saliency mod-

978-1-4799-6683-7/15 $31.00 © 2015 IEEE
DOI 10.1109/WACV.2015.135

els in the frequency domain shown in Figure 1. The phase and amplitude spectrums of an image has been utilized for frequency domain saliency models. Most existing models retain the original phase spectrum and only modify the amplitude spectrum to generate saliency maps. We propose four frequency domain visual saliency models based on new spectrum processing methods and an entropy-based saliency map selection approach. Our models are the Gamma Corrected Spectrum (GCS) model, the Gamma Corrected Log Spectrum (GCLS) model, the Gaussian Filtered Spectrum (GFS) model, and the Gaussian Filtered Log Spectrum (GFLS) model. A set of saliency map candidates are generated by inverse transform of a set of modified spectrums. An entropy-based approach is used to select a "good" saliency map by minimizing the entropy among the set of saliency map candidates. A group of extended models are also described using various color spaces. A visual saliency model should be capable of providing accurate prediction of human eye fixation/tracking data using eye fixation image datasets. We did a comprehensive evaluation of four of our best extended models (GCS-FT-Lab, GCLS-FT-Lab, GFS-FT-Lab, and GFLS-HFT-IRGBY) by comparing them with 10 state-of-the-art saliency models using two standard image datasets.

2. Frequency Domain Visual Saliency Models

The existing frequency domain based visual saliency models described above are of two types: (1) Some frequency domain models generate saliency maps using color channel images separately. They process the spectrum of each color channel image independently and then fuse the individual saliency maps into the final map (*e.g.* [13, 17]). (2) The other frequency domain models generate saliency maps using a composite image representation. They usually merge color channel images into a quaternion image and then use the Hypercomplex Fourier Transform (HFT) [8, 9] to obtain the quaternion spectrum for processing (*e.g.* [14, 22]). Note that the ideas of separate and composite processes have been alternatively presented in existing frequency domain models and we generalize frequency domain visual saliency models in Figure 1 when using different spatial domain and frequency domain operations.

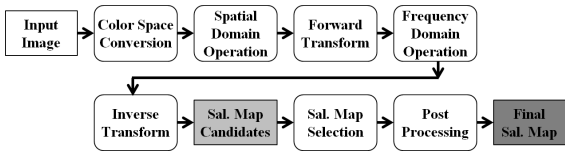


Figure 1. The generalization of frequency domain visual saliency models.

2.1. Color Spaces

Most visual saliency models are based on color spaces. A color space is a geometric representation of colors in a space, usually three dimensions that refers to three color channels. The **RGB** color space is an well-known additive color space based using three color primaries, *i.e.* red, green, and blue [27]. RGB color spaces are often used for image/video capture, representation, and display. The **Lab** ($CIE\ L^*a^*b^*$) color space is widely used because it represents human perceptual uniformity for color difference measurement [11]. The L^* component reflects human perception of lightness while the a^* and b^* components approximate the human chromatic opponent system. The **IRGBY** opponent color space is used because there exists a color double-opponent system in the human visual cortex for the red/green, green/red, blue/yellow, and yellow/blue color pairs [10]. Let r, g, and b denote the red, green, and blue color primaries, four color features are first generated as follows (negative values are set to zero). Four color features are first determined by $R = r - (g + b)/2$ for red, $G = g - (r + b)/2$ for green, $B = b - (r + g)/2$ for blue, and $Y = (r + g)/2 - |r - g|/2 - b$ for yellow (negative values are set to zero). The IRGBY opponent color space is formed by the intensity channel ($\mathcal{I} = (r + g + b)/3$) and the two opponent color channels ($\mathcal{RG} = R - G$ and $\mathcal{BY} = B - Y$).

3. Proposed Visual Saliency Models

The phase spectrum contains important saliency information that indicates where the "proto-objects" or salient regions are located in the spatial domain [13, 17]. The amplitude spectrum also contains both saliency (distinct patterns) and non-saliency (repeated patterns) information. The sharp peaks or spikes in the amplitude spectrum correspond to non-saliency regions which should be suppressed for saliency detection [22]. Most existing frequency domain models only process the amplitude spectrum while leaving the phase spectrum unchanged when generating saliency maps. Given an image $f(x, y)$, it is "transformed" into its frequency domain representation and denoted as $\mathcal{F}(u, v) = T[f(x, y)]$. The amplitude spectrum is defined as $\mathcal{A}(u, v) = |\mathcal{F}(u, v)|$ and the phase spectrum is defined as $\mathcal{P}(u, v) = angle(\mathcal{F}(u, v))$. If necessary the log amplitude spectrum may be also be defined as $\mathcal{L}(u, v) = log_e(\mathcal{A}(u, v)) = log_e(|\mathcal{F}(u, v)|)$. Note that the "transform" used for each visual saliency model depends on which saliency model family is used. The inverse transform can be written as follows:

$$f(x, y) = T^{-1}[\mathcal{F}(u, v)], \quad (1)$$

$$\Leftrightarrow f(x, y) = T^{-1}[\mathcal{A}(u, v) \cdot exp(i \cdot \mathcal{P}(u, v))], \quad (2)$$

$$\Leftrightarrow f(x, y) = T^{-1}[exp(\mathcal{L}(u, v) + i \cdot \mathcal{P}(u, v))]. \quad (3)$$

The saliency map can be considered as a probability map whose values range from 0 to 1. The higher salient regions would be assigned larger probability values. We use the entropy of the saliency map to select a "good" saliency map with the lowest fragmentation and randomness. A saliency map $S'(x, y, k')$ is selected by minimizing the entropy $\mathcal{H}(S(x, y, k))$ among the group of saliency map candidates.

$$k' = \arg\min_{k}\{\mathcal{H}(S(x, y, k))\}, \tag{4}$$

$$\mathcal{H}(S) = -\sum_{g=1}^{L} P_{S,g} log_2(P_{S,g}), \tag{5}$$

$$P_{S,g} = \frac{C_{S,g}}{M \times N}, \tag{6}$$

where $\mathcal{H}(S(x, y, k))$ is the entropy of each saliency map candidate $S(x, y, k)$. The priori probability $P_{S,g}$ is defined by the total number of occurrences $C_{S,g}$ (pixel counts in histogram) of the saliency level g divided by the total number of pixels $M \times N$ of the saliency map $S(x, y, k)$.

The final saliency map is generated after post-processing steps [5]. As described in [18], usually each element in the saliency map is squared individually and then the saliency map is saliency map is convoluted with a Gaussian burring kernel $b_{opt}(x, y)$ with an optimal sigma σ_{opt} determined by experiments.

$$S''(x, y, k') = b_{opt}(x, y) \star \|S'(x, y, k')\|^2, \tag{7}$$

where $b_{opt}(x, y)$ is a Gaussian burring kernel with optimal sigma σ_{opt}, $\star$ denotes the convolution operation and $\|\cdot\|^2$ denotes the square of each element individually.

3.1. Visual Saliency Model Using Gamma Corrected Spectrum (GCS)

Gamma correction is a nonlinear operation used to modify the luminance or tristimulus values in an image display system [24]. It is defined by two reversible power functions, $V_{out} = (V_{in})^{\gamma}$ and $V_{in} = (V_{out})^{\frac{1}{\gamma}}$, where V_{out} and V_{in} are the input and output values. Under common illumination conditions, the human visual systems follows an approximate power function, namely the psychophysical power law, developed by Stanley S. Stevens [26]. Gamma correction is used to compensate for the human visual system, in order to maximize the use of the bits or bandwidth according to how humans perceive light or color [24].

We propose a visual saliency model using the Gamma Corrected Spectrum (GCS). A set of gamma corrections with different gamma values γ_k are utilized to modify the amplitude spectrum while keeping the phase spectrum unchanged. Saliency map candidates $S(x, y, k)$ can be constructed by the inverse transform of the gamma corrected amplitude spectrums $\mathcal{A}_{GCS}(u, v, k)$ with the original phase spectrum $\mathcal{P}(u, v)$.

$$\mathcal{A}_{GCS}(u, v, k) = (\mathcal{A}(u, v))^{\gamma_k}, \tag{8}$$

$$S(x, y, k) = T^{-1}[(\mathcal{A}(u, v))^{\gamma_k} \cdot exp(i \cdot \mathcal{P}(u, v))], \tag{9}$$

$$S(x, y, k) = T^{-1}[exp(\mathcal{L}(u, v) \cdot \gamma_k + i \cdot \mathcal{P}(u, v))], \tag{10}$$

where k is an index $k = \{0, \ldots, K\}$ and $\gamma_k = \frac{k}{16}$. K is determined by the largest dimension of the size of the saliency map, $K = \lfloor log_4(max(H, W)) \rfloor + 1$, where W and H are the width and height of the saliency map. For example, if the size of the saliency map is 64×48, $K = 4$, $k = \{0, 1, 2, 3, 4\}$, and $\gamma_k = \{0, \frac{1}{16}, \frac{1}{8}, \frac{1}{4}, \frac{1}{2}\}$. An output saliency map $S'(x, y, k')$ is selected by minimizing the entropy $\mathcal{H}(S(x, y, k))$ among the set of candidates using the Equation (4) and (5). The final GCS saliency map is obtained by Equation (7).

3.2. Visual Saliency Model Using Gamma Corrected Log Spectrum (GCLS)

Following our Gamma Corrected Spectrum (GCS) model, we also investigated a visual saliency model using the Gamma Corrected Log Spectrum (GCLS). A set of gamma corrections with different gamma values γ_k are used to modify the log amplitude spectrum while retaining the phase spectrum unchanged. For convenience, we only describe the main steps in the following equations.

$$\mathcal{L}_{GCLS}(u, v, k) = (\mathcal{L}(u, v))^{\gamma_k}, \tag{11}$$

$$S(x, y, k) = T^{-1}[exp((\mathcal{L}(u, v))^{\gamma_k} + i \cdot \mathcal{P}(u, v))]. \tag{12}$$

We use the same parameter settings as the GCS model and an output saliency map $S'(x, y, k')$ is selected by the same selection approach using the Equation (4) and (5). The final GCLS saliency map is obtained by Equation (7).

3.3. Visual Saliency Model Using Gaussian Filtered Spectrum (GFS)

Inspired by the related work in [22], we propose another visual saliency model using the Gaussian Filtered Spectrum (GFS). A set of Gaussian filters $GF(u, v, k)$ with various standard deviations σ_k are used to filter the amplitude spectrum while retaining the phase spectrum unchanged. Saliency map candidates $S(x, y, k)$ can be constructed by the inverse transform of the Gaussian filtered amplitude spectrums $\mathcal{A}_{GFS}(u, v, k)$ with the original phase spectrum $\mathcal{P}(u, v)$.

$$\mathcal{A}_{GFS}(u, v, k) = \mathcal{A}(u, v) \star GF(u, v, k), \tag{13}$$

$$S(x, y, k) = T^{-1}[\mathcal{A}_{GFS}(u, v, k) \cdot exp(i \cdot \mathcal{P}(u, v))], \tag{14}$$

where k is an index $k = \{1, \ldots, K\}$ and $\sigma_k = 4^{k-1}$. K is determined by the largest dimension of the size of the

saliency map, $K = \lfloor log_4(max(H, W)) \rfloor + 2$, where W and H are the width and height of the saliency map. For example, if the size of the saliency map is 64×48, $K = 5$, $k = \{1, 2, 3, 4, 5\}$, and $\sigma_k = \{1, 4, 16, 64, 256\}$. An output saliency map $S'(x, y, k')$ is selected by minimizing the entropy $\mathcal{H}(S(x, y, k))$ among the set of candidates using the same Equation (4) and (5). The final GFS saliency map is obtained by Equation (7).

3.4. Visual Saliency Model Using Gaussian Filtered Log Spectrum (GFLS)

Following our Gaussian Filtered Spectrum (GFS) model, we also investigated a model using the Gaussian Filtered Log Spectrum (GFLS). A set of Gaussian filters $GF(u, v, k)$ with various standard deviations σ_k are used to filter the log amplitude spectrum while keeping the phase spectrum unchanged. For convenience, we only describe the main steps in the following equations.

$$\mathcal{L}_{GFLS}(u, v, k) = \mathcal{L}(u, v) \star GF(u, v, k), \quad (15)$$

$$S(x, y, k) = T^{-1}[exp(\mathcal{L}_{GFLS}(u, v, k) + i \cdot \mathcal{P}(u, v))]. \quad (16)$$

We use the same parameter settings as the GFS model and an output saliency map $S'(x, y, k')$ is selected by the same selection approach using the Equation (4) and (5). The final GFLS saliency map is obtained by Equation (7).

We also investigated several other models by extending our proposed models above to fit both separate and composite model families and various color spaces. The **naming convention of the extended models** is defined as "**A-B-C**", where **A** represents a proposed model's abbreviation (*i.e.* GCS, GCLS, GFS, and GFLS) or an existing model's abbreviation (*i.e.* IS and SSA); **B** represents a specific transform used for an extended model (*i.e.* FT and HFT); **C** represents a particular color space used for an extended model, including Lab, IRGBY, and RGB color spaces.

4. Experimental Results

Previous studies have shown that good visual saliency models are capable of providing significantly accurate prediction of human eye fixation on natural images in free viewing [12, 29, 4, 5]. Our experiments use our proposed visual saliency models and some state-of-the-art models to predict human eye fixation on two standard image datasets. (1) The Bruce and Tsotsos (BT) dataset [6] is the most widely used dataset for comparing visual saliency models. It contains 120 color images with resolution of 681x511 pixels from indoor and outdoor scenes and the eye fixation data is based on 20 subjects. (2) The Li *et al.*'s (Li) dataset [22] is a new dataset containing 235 color images with resolution of 640x480 pixels in six categories. We used the original MATLAB implementation of the state-of-the-art models with the default settings and the recommended saliency map's resolution. In our experiments, we kept all post-processing settings of each model as original except the blurring/smoothing parameter. Because blurring/smoothing the resulting saliency maps is an important factor for fair comparison [17], we tuned each model to achieve its best performance on each standard image datasets by searching and selecting the optimal parameter of Gaussian blurring/smoothing. We blurred the saliency map of each model by convoluting them with a series of Gaussian kernels with different standard deviations σ (from 0.005 to 0.1 in steps of 0.005) in terms of the largest dimension of an image. We implemented all of our extended models in MATLAB and set the saliency map's resolution to 64×48 pixels in all the experiments. The experiments were executed on a backend server with four quad-core 3.2GHz CPU and 32GB RAM.

We adopted the shuffled AUC (sAUC) score [28, 32] as the fair evaluation measure and developed an enhanced evaluation tool to compare various visual saliency models based on an existing benchmark [17]. The sAUC is the shuffled Area Under the Receiver Operating Characteristics (ROC) Curve. Experimental results indicate that the Lab and IRGBY color spaces work better with the FT-based models in the separate model family and that the IRGBY color space works better with the HFT-based models in the composite model family. Regarding the three color spaces, Lab-based and IRGBY-based extended models are generally better than RGB-based extended models in predicting human eye fixation. For the four groups of extended saliency models, we selected the best extended model in each group based on our experiments, *i.e.* GCS-FT-Lab, GCLS-FT-Lab, GFS-FT-Lab, and GFLS-HFT-IRGBY.

We did a comprehensive evaluation of our four best extended saliency models (GCS-FT-Lab, GCLS-FT-Lab, GFS-FT-Lab, and GFLS-HFT-IRGBY) by comparing them with 10 state-of-the-art saliency models using two standard image datasets. The 10 models used in this experiment are: SBVA(Itti) [20], AIM [6], FTSD [1], PFDN [3], SR [18], PFT [13], PQFT [14], QDCT [25], IS-DCT-Lab [17], SSA-HFT-IRGBY [22] The SBVA(Itti), AIM, and FTSD are spatial domain models while the PFDN, SR, PFT, PQFT, IS-DCT-Lab, SSA-HFT-IRGBY and the four best extended models are all frequency domain models. Figure 2 and Figure 4 demonstrate the sAUC score of each model based on the two standard image datasets. Figure 3 and Figure 5 illustrate the average execution time per image of each model. Note that the time axis less than 0.3 second uses a linear scale greater than 0.3 second is a non-linear scale based on the largest value of the average execution time per image.

The rank of each saliency model, the maximum sAUC score, and the associated optimal Gaussian standard deviation σ_{opt} are shown in Table 1. The results show that our four best extended models are generally better than most state-of-the-art models. They are ranked in the top 6 out

Table 1. The rank of each saliency model, maximum sAUC score, and the associated Gaussian σ_{opt} (in image largest dimension).

Image Dataset	SBVA [20]	AIM [6]	FTSD [1]	PFDN [3]	SR [18]	PFT [13]	PQFT [14]	QDCT [25]	IS-DCT [17]	SSA-HFT [22]	GCS-FT -Lab	GCLS-FT -Lab	GFS-FT -Lab	GFLS-HFT -IRGBY
BT [6] Rank	13	8	14	7	11	11	10	5	**3**	9	**2**	**1**	**4**	6
Max sAUC	0.6453	0.6956	0.5883	0.7015	0.6898	0.6898	0.6906	0.7041	0.7111	0.6932	0.7135	0.7138	0.7087	0.7030
σ_{opt}	0.030	0.030	0.040	0.045	0.040	0.040	0.040	0.040	0.040	0.040	0.045	0.040	0.045	0.040
Li [22] Rank	13	9	14	10	11	12	8	7	6	**4**	**1**	**2**	5	**2**
Max sAUC	0.6525	0.6727	0.6186	0.6724	0.6649	0.6639	0.6742	0.6757	0.6758	0.6795	0.6805	0.6803	0.6780	0.6803
σ_{opt}	0.035	0.040	0.045	0.050	0.055	0.050	0.050	0.050	0.050	0.045	0.050	0.050	0.050	0.050

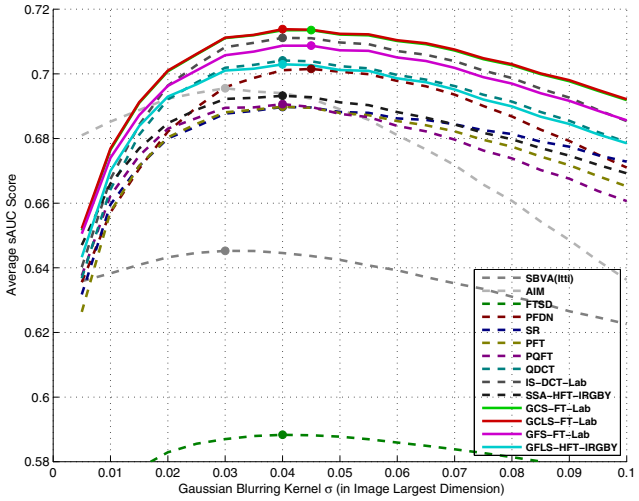


Figure 2. The sAUC score of each model (BT dataset).

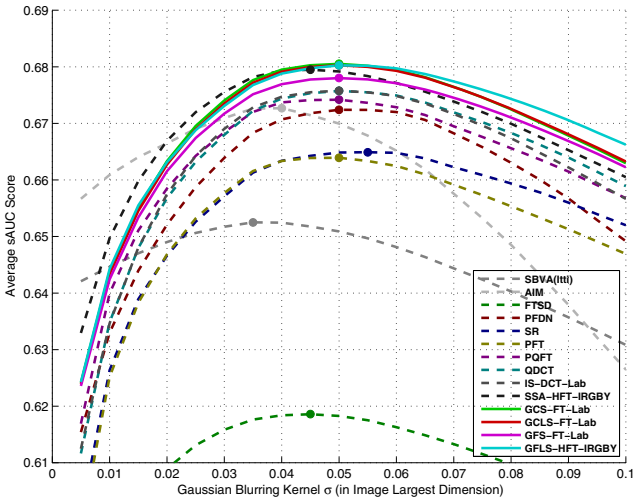


Figure 4. The sAUC score of each model (Li dataset).

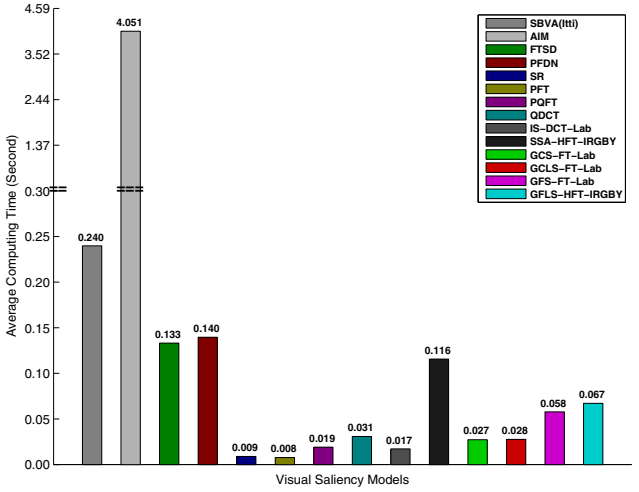


Figure 3. The average execution time of each model (BT dataset).

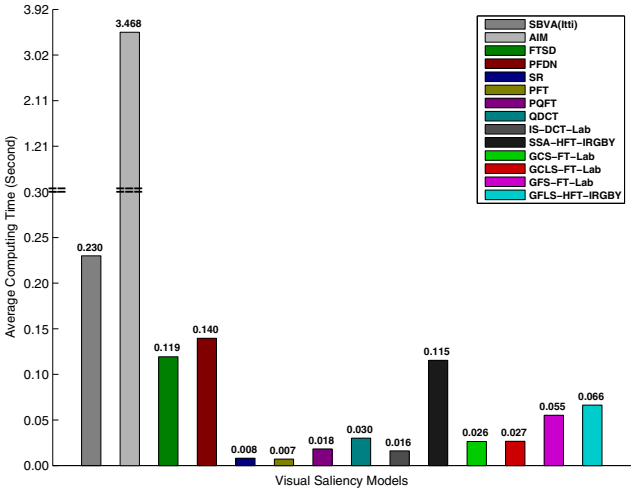


Figure 5. The average execution time of each model (Li dataset).

of all the 14 models for the BT dataset and the top 5 out of all the 14 models for the Li dataset. Among the existing models, IS-DCT-Lab [17], SSA-HFT-IRGBY [22], and QDCT [25] models performed better than the rest while FTSD [1], SBVA(Itti) [20], PFT [13], and SR [18] models are ranked at the bottom. The GCS-FT-Lab and GCLS-FT-Lab models are also the two most accurate ones in predicting eye fixation. Since the GCS-FT-Lab model has lower complexity and is more efficient than the GCLS-FT-Lab model, we consider the GCS-FT-Lab model as the best of all the 14 models. Due to space limitations, one can see more saliency map examples in the supplementary material.

5. Conclusions

In this paper we investigated low-complexity visual saliency models using spectral analysis approaches. We proposed four visual saliency models based on new spectrum processing methods and an entropy-based saliency map selection approach. A group of saliency models that extends each of the proposed models was also described using various color spaces. Experimental results show that the four best extended models are more accurate and efficient than some state-of-the-art models in predicting eye fixation on standard image datasets.

Acknowledgement: This work was supported by the Visual Analytics for Command, Control, and Interoperability Environments (VACCINE) Center of the U.S. Department of Homeland Security (DHS) under Award Number 2009-ST-061-CI000.

References

[1] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1597–1604, June 2009. Miami, FL, USA.

[2] S. Avidan and A. Shamir. Seam carving for content-aware image resizing. *ACM Transactions on Graphics*, 26(3):1–9, July 2007.

[3] P. Bian and L. Zhang. Visual saliency: A biologically plausible contourlet-like frequency domain approach. *Cognitive Neurodynamics*, 4(3):189–198, Sep. 2010.

[4] A. Borji and L. Itti. State-of-the-art in visual attention modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):185–207, Jan. 2013.

[5] A. Borji, D. N. Sihite, and L. Itti. Quantitative analysis of human-model agreement in visual saliency modeling: A comparative study. *IEEE Transactions on Image Processing*, 22(1):55–69, Jan. 2013.

[6] N. D. B. Bruce and J. K. Tsotsos. Saliency, attention, and visual search: An information theoretic approach. *Journal of Vision*, 9(3):1–24, Mar. 2009.

[7] C. Christopoulos, A. Skodras, and T. Ebrahimi. The JPEG2000 still image coding system: An overview. *IEEE Transactions on Consumer Electronics*, 46(4):1103–1127, Nov. 2000.

[8] T. A. Ell. Quaternion-Fourier transforms for analysis of two-dimensional linear time-invariant partial differential systems. *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 2:1830–1841, Dec. 1993. San Antonio, TX, USA.

[9] T. A. Ell and S. J. Sangwine. Hypercomplex Fourier transforms of color images. *IEEE Transactions on Image Processing*, 16(1):22–35, Jan. 2007.

[10] S. Engel, X. Zhang, and B. Wandell. Colour tuning in human visual cortex measured with functional magnetic resonance imaging. *Nature*, 388(6637):68–71, July 1997.

[11] M. D. Fairchild. *Color Appearance Models*. Wiley-IS&T, Chichester, UK, 3 edition, 2013.

[12] S. Frintrop, E. Rome, and H. I. Christensen. Computational visual attention systems and their cognitive foundations: A survey. *ACM Transactions on Applied Perception*, 7(1):6:1–6:39, Jan. 2010.

[13] C. Guo, Q. Ma, and L. Zhang. Spatio-temporal saliency detection using phase spectrum of quaternion Fourier transform. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2008. Anchorage, AK, USA.

[14] C. Guo and L. Zhang. A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression. *IEEE Transactions on Image Processing*, 19(1):185–198, Jan. 2010.

[15] J. Han, K. N. Ngan, M. Li, and H. Zhang. Unsupervised extraction of visual attention objects in color images. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(1):141–145, Jan. 2006.

[16] J. Harel, C. Koch, and P. Perona. Graph-based visual saliency. *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, pages 545–552, Dec. 2006. Vancouver, BC, Canada.

[17] X. Hou, J. Harel, and C. Koch. Image signature: Highlighting sparse salient regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(1):194–201, Jan. 2012.

[18] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2007. Minneapolis, MN, USA.

[19] L. Itti. Automatic foveation for video compression using a neurobiological model of visual attention. *IEEE Transactions on Image Processing*, 13(10):1304–1318, Oct. 2004.

[20] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, Nov. 1998.

[21] C. Koch and S. Ullman. Shifts in selective visual attention: Towards the underlying neural circuitry. *Human Neurobiology*, 4(4):219–227, Apr. 1985.

[22] J. Li, M. D. Levine, X. An, X. Xu, and H. He. Visual saliency based on scale-space analysis in the frequency domain. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(4):996–1010, Apr. 2013.

[23] A. Mogelmose, M. M. Trivedi, and T. B. Moeslund. Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey. *IEEE Transactions on Intelligent Transportation Systems*, 13(4):1484–1497, Dec. 2012.

[24] C. A. Poynton. Rehabilitation of gamma. *Proceedings of the SPIE conference on Human Vision and Electronic Imaging III*, 3299:232–249, Jan. 1998. San Jose, CA, USA.

[25] B. Schauerte and R. Stiefelhagen. Predicting human gaze using quaternion DCT image signature saliency and face detection. *Proceedings of the IEEE Workshop on Applications of Computer Vision (WACV)*, pages 137–144, Jan. 2012. Breckenridge, CO, USA.

[26] S. S. Stevens. On the psychophysical law. *Psychological Review*, 64(3):153–181, May 1957.

[27] S. Süsstrunk, R. Buckley, and S. Swen. Standard RGB color spaces. *Proceedings of the Color Imaging Conference (CIC): Color Science, Systems, and Applications*, pages 127–134, Nov. 1999. Scottsdale, AZ, USA.

[28] B. W. Tatler, R. J. Baddeley, and I. D. Gilchrist. Visual correlates of fixation selection: Effects of scale and time. *Vision Research*, 45(5):643–659, Mar. 2005.

[29] A. Toet. Computational versus psychophysical bottom-up image saliency: A comparative evaluation study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(11):2131–2146, Nov. 2011.

[30] A. M. Treisman and G. Gelade. A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97–136, Jan. 1980.

[31] D. Walther and C. Koch. Modeling attention to salient proto-objects. *Neural Networks*, 19(9):1395–1407, Nov. 2006.

[32] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell. SUN: A bayesian framework for saliency using natural statistics. *Journal of Vision*, 8(7):1–20, Dec. 2008.

Analyzing High-dimensional Multivariate Network Links with Integrated Anomaly Detection, Highlighting and Exploration

Sungahnn Ko, Shehzad Afzal, Simon Walton, Yang Yang, Junghoon Chae, Abish Malik,
Yun Jang, Min Chen and David Ebert, *Fellow, IEEE*

Abstract—This paper focuses on the integration of a family of visual analytics techniques for analyzing high-dimensional, multivariate network data that features spatial and temporal information, network connections, and a variety of other categorical and numerical data types. Such data types are commonly encountered in transportation, shipping, and logistics industries. Due to the scale and complexity of the data, it is essential to integrate techniques for data analysis, visualization, and exploration. We present new visual representations, *Petal* and *Thread*, to effectively present many-to-many network data including multi-attribute vectors. In addition, we deploy an information-theoretic model for anomaly detection across varying dimensions, displaying highlighted anomalies in a visually consistent manner, as well as supporting a managed process of exploration. Lastly, we evaluate the proposed methodology through data exploration and an empirical study.

1 Introduction

The recent trend of increasing size, complexity, and variety in datasets (e.g., spatial, temporal, quantitative, qualitative, network data) makes analysis and decisions from these data more challenging, often called the *big data* problem [21, 32, 37]. One very challenging type of big data is multivariate network data, especially when there are multivariate values for both nodes and links. For example, transportation, shipping, logistics, commerce, trading, electricity and communication industries [7, 41] have many connected operational locations where multiple variables describe each location's operations. With flight delay network data, various multivariate operational aspects are considered simultaneously: types of delay, patterns based on airport location, trends in time, and relationships among the airports. To reduce the analysts' information overload and to enable effective planning, analysis and decision making, an interactive visual exploration and analysis environment is needed as traditional machine learning and big data analytics alone are insufficient [10].

While, various systems and techniques for network visualization have been proposed [19], few support analyzing both multivariate network data (e.g., [39] and [25]) and map-based spatial network data (e.g., [17] and [7]). There still remains a gap in effective multivariate spatial network data exploration and analysis to efficiently answer challenging questions such as the following: What are the patterns in multivariate variables on a node or among node-node pairs? Are the patterns relevant to specific regions and times? Is there any seasonality in the patterns? Can we verify the patterns on a map? Which network nodes and links could be anomalous?

In this work, we fill this gap by integrating a family of visual analytics techniques for exploring and analyzing such complex data. We employ multiple linked views [30] (see Fig. 1), two new multivariate visualization techniques, *petals* and *threads*, and an information-theoretic analytical backend engine for aggregate-level and detail-level network analysis.

Petals and *threads* efficiently present a simplified representation of many-to-many networks where multi-attribute vectors represent the size of attributes in different directions. Specifically, *petals* represent an aggregated summary view of directional data (Fig. 3) and *threads* encode multiple variables of links (Fig. 2). An information-theoretic model provides our analytical engine the ability to highlight anomalies in the data. The anomaly detection can be dynamically configured based on new contextual requirements that usually result from user-generated hypotheses stimulated from visualization and exploration of data. The analytical method provides visualization with additional warning signals and enables users to prioritize their exploration strategy.

The contributions of our work in the multivariate spatiotemporal network visualization and analysis domain are 1) designing *petals* and *threads* for high-dimensional multivariate network link analysis, 2) evaluating *petals* and *threads* with a user study, 3) designing and implementing a visual analytics system using multiple coordinated views, 4) integrating an information-theoretic anomaly detection method in the interactive visualization analysis process, and 5) exploring complex data (e.g., flight delay network) to illustrate the use and potential of our designs in the multiple-coordinated views.

Our system can be applied to exploration of any multivariate spatiotemporal, network link data such as transportation, shipping, logistics, commerce, trading, communication industries (e.g., AT&T communication network data [7] and electric power grid data [41]).

2 Related Work

While the research topics in network visualization are as numerous as the visualizations themselves [19], in this work, we consider network visualization techniques and tools that are pertinent to multivariate geospatial network data. For multivariate network visualization research, Wattenberg [39] has designed PivotGraph, a software tool focusing on the relationships between node attributes and connections of multivariate graphs on a grid layout. Ploceus [25] enables multi-dimensional and multi-level network-based visual analysis on tabular data while Honeycomb [38] focuses on scalability (e.g., millions of connections) using a matrix representation that is also incorporated in our pixel matrix view. For geospatial network visualization, Guo [17] has developed an integrated, interactive visualization framework that visualizes major flow structures and multivariate relations at the same time. SeeNet [7] visualizes geospatial network data in a communication industry; however, its visualization focuses on univariate data. In contrast to the previous work, our system allows users to analyze all combinations of spatial, temporal, multivariate, and network characteristics simultaneously. Herman et al. [19] surveyed other network visualization techniques beyond our paper's scope.

In order to visualize multivariate data, and to display the maximum amount of data relative to the available screen space, a pixel-based visualization was developed by Keim et al. [20]. In the pixel-based visualization, each data element is assigned to a pixel, and a predefined color map is used to shade the pixel to represent the range of

- *Sungahn Ko, Shehzad Afzal, Yang Yang, Junghoon Chae, Abish Malik and David S. Ebert are with Purdue University in West Lafayette, IN, USA. E-mail: {ko, safzal, yang260, jchae, amalik, ebertd}@purdue.edu.*
- *Yun Jang is with Sejong University in Seoul, Korea. E-mail: {jangy}@sejong.edu.*
- *Simon Walton and Min Chen are with Oxford University in Oxford, UK. E-mail: {simon.walton, min.chen}@oerc.ox.ac.uk.*

Submitted to IEEE VAST 2014. Do not redistribute.

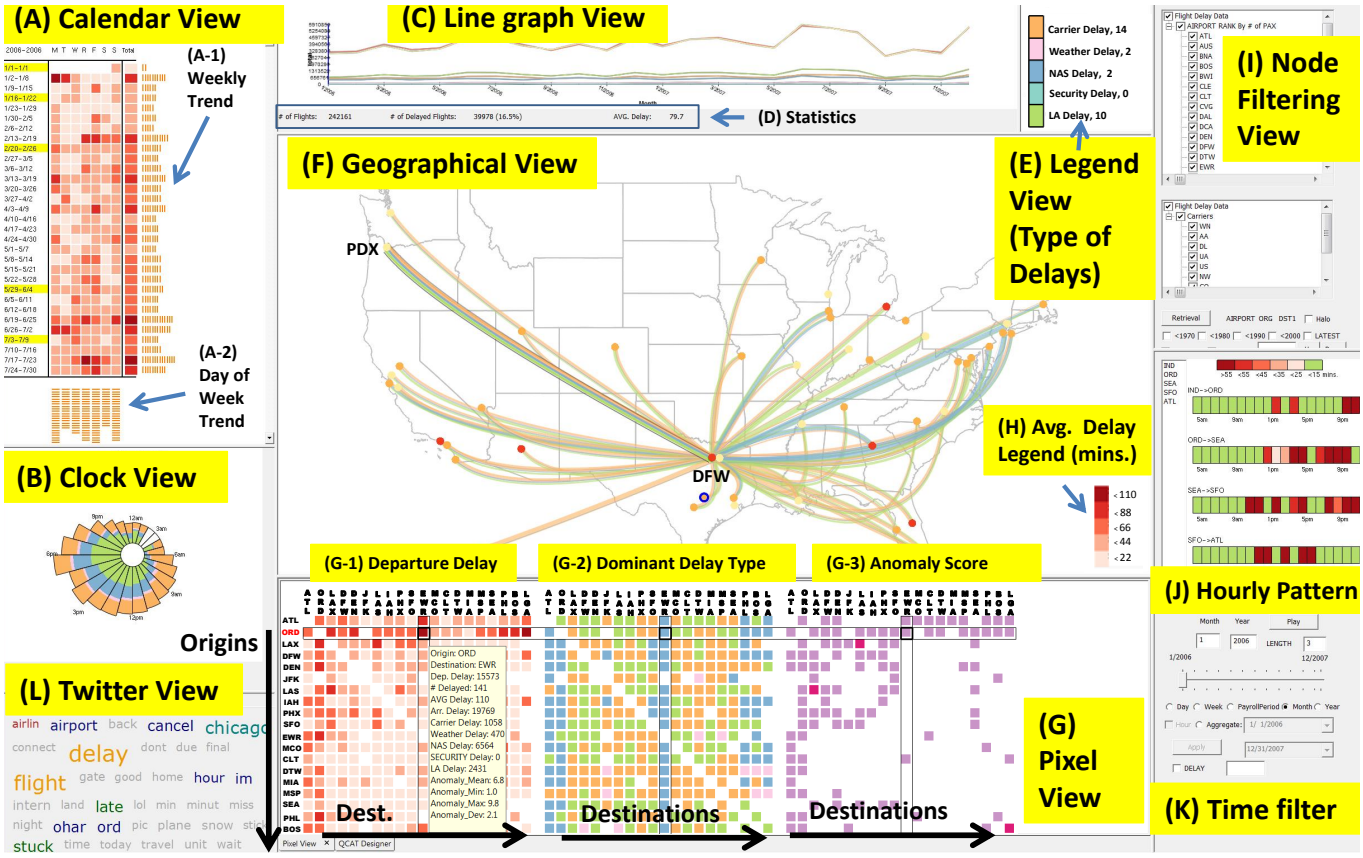


Fig. 1. Our system consists of multiple coordinated and linked views: (A) Calendar view, (B) Clock view, (C) Line graph view, (D) Statistics, (E) Legend view for displaying types of delays, (F) Geographical view, (G) Pixel view, (H) Legend view for delay type and time, (I) Node filter, (J) Pattern on itinerary view, (K) Time and aggregation filter, and (L) Twitter tag cloud view. In the (H) legend, the darker the red, the longer the average delay is. A route from Dallas (DFW) to Portland (PDX) is specified in (F), and the top 20 airports in terms of delays are visualized in (G) for explanation.

the data attribute. Thus, the amount of information in the visualization is theoretically limited only by the resolution of the screen. Borgo et al. [8] present how the usability of the pixel-based visualization varies across different tasks and block resolutions. Oelke et al. [29] study visual boosting techniques for pixel-based visualization such as halos and distortion. Ziegler et al. [43] present how pixel-based visualizations help analysts gain insight for long-term investments while Ko et al. [22] demonstrate the effectiveness of pixel-based visualization in analyzing corporate competitive advantages. Our system incorporates this pixel-based visualization not only to visualize as much data as possible, but also to describe origin-destination network status because the conventional layout of placing pixels side-by-side naturally builds a node-link network.

To help users visually explore multivariate data, many systems have been developed in research and commercial areas [42] (e.g., Spotfire [5], QlikView [3], and Tableau [4]). Common among these systems is that they make extensive use of interactive techniques for brushing, linking, zooming, and filtering to refine the user's queries. Of the systems, Tableau [4], which has become popular due to its flexible operation, allows analysts to easily access and effectively analyze their data [42]. Although multivariate and time-series data analysis is possible in the tool, comparison among multivariate, spatial-temporal, and network-based attributes with geographical components is not well supported by Tableau. In our system, all attributes and characteristics in the data are incorporated and visualized using multiple linked views for simultaneous comparisons. For visualizing multivariate data, Duffy et al. [13] use a glyph encoding some 20 variables while Scheepens et al. [33] focus on a method for reducing visual clutter and occlusion among glyphs.

Analysis of spatiotemporal social media data can have a significant impact to increase situational awareness and provide insights for investigations. Sakaki et al. [31] introduce a natural disaster alert system for earthquake epicenter estimation by analyzing Twitter messages as virtual sensors. Scatterblogs [9,36] is a scalable system enabling analysts to find quantitative information and to detect spatiotemporal anomalies within a large volume of geo-located microblog messages. Chae et al. [11] propose an interactive social media data analysis and visualization system for anomalous circumstance detections as well as examinations of abnormal topics and events from various social media data sources. In our user study, we use Twitter tags to help analysts find any correlation between data and public reactions.

Lee and Ziang [24] provide an overview of using information-theoretical measures for anomaly detection, including entropy, conditional entropy, information gain, and information cost. A number of case studies are also provided in the domain of network security. Chandola et al. provided a comprehensive survey on methods for anomaly detection [12]. Arackaparambil et al. [6] use information theory to monitor network streams for anomalies in network traffic, and to explore the challenges of providing a scalable implementation using a distributed approach to computing entropy and conditional entropy. Kopylova et al. [23] investigate the use of mutual information in network traffic anomaly detection using Rényi entropy rather than the traditional Shannon entropy measure.

3 Multivariate Network Visualization

To effectively reveal as many aspects of the data characteristics as possible, we explore the data in a series of linked visualizations. Fig. 1illustrates how our system provides a comprehensive multivariate network information in multiple linked views. For illustration, we use a flight delay network dataset [1] as an example of multivariate geospatial network data, but any multivariate network data can be populated into our system. Multivariate network information is provided in the

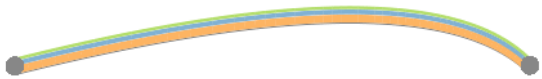


Fig. 2. *Thread* example, showing a single link with multiple *threads*. Width of each *thread* within a link is adjusted based on the contribution of each variable. Contribution of each variable in this example is as follows: Variable 1 (Orange) = 0.5, Variable 2 (Blue) = 0.3, Variable 3 (Green) = 0.2.

geographical view (F) where any operational variable can be used for coloring the node (e.g., anomaly score). The user can explore the data in either a pixel view or a parallel coordinate view (G). Note that (G) has two tab views at the bottom, and a parallel coordinate view example in (G) is shown in Fig. 8. Similarly, time-varying variables (e.g., delays) are presented in different linked visualizations for efficient exploration in the calendar view (A), clock view (B), and line graph view (C). The Twitter view (L) presents important tags with different font sizes and colors. Hourly (delay) patterns can be found in (J) on a series of linked nodes. For example, when a user plans to travel in a certain order (e.g., IND, ORD, SEA, SFO, and ATL), the user can easily find when severe delays are caused in each origin–destination pair. In the system, the line graph view presents temporally aggregated data (e.g., weekly, monthly, yearly). The parallel coordinate view (discussed later in Section 5.2) can be be used to explore the attributes and their value distributions, as well as designing and selecting Query Conditional ATtributes(QCATs, discussed in Section 4) for anomaly detection. Based on characteristics of the data, perceptually appropriate color maps are chosen from both sequential and qualitative color maps from ColorBrewer [18].

3.1 Spatial Multivariate Network Visualization

Unfortunately, a barrier exists in analyzing multivariate network data because visual clutter and complexity often occur in visualizing multiple variables for a node with multiple links between nodes in the map. To reduce such clutter and complexity in the analysis, we design *threads* (see Fig. 2) and *petals* (see Fig. 3) for exploring multivariate link network data. *Threads* connect an origin to each destination and visualize multiple link variables. Because visual clutter around the origin is often generated by link visualization and our *threads*, we also design *petals* to present aggregated and simplified many-to-many network link data. *Threads* and *petals* are designed based on the following requirements for the visualization:

R1 A visualization should present multiple variables describing the relationships between an origin and multiple destination nodes on the map. Here, users should be able to see an overview of the multivariate relationships and discern at least the largest variable in the visualization for both one-to-one and one-to-many relationships.

R2 The visualization should provide simplified one-to-many multivariate spatial networks with minimum visual clutter. Use of node rearrangement techniques (e.g., force-based model algorithm [28]) is not allowed to maintain geospatial semantic meanings.

R3 Users should be able to discern in the visualization for R2 which one-to-many network has the largest aggregate value and which variable has the largest contribution for the largest aggregate value of the one-to-many network.

R4 Multiple variables describing the statistics for a node should be visually presented.

For goal R1, we design *threads*, and for goals R2–R4 we design *petals*. In the following sections, we explain their visual representations in detail.

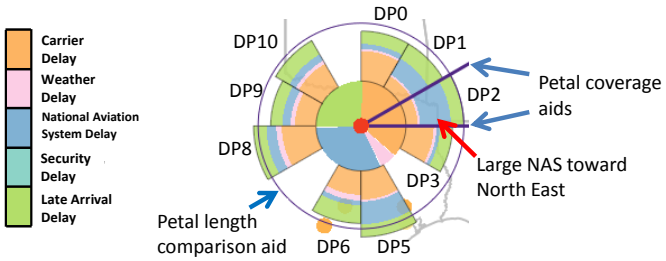


Fig. 3. To show the *petal* coverage including destinations, *petal* coverage guide lines are provided. For comparison of *petal* lengths, equal-radius circles are drawn on all *petals* as shown. The radius of the circles is the length of the *petal* where a user's mouse is hovering.

3.1.1 Thread Visual Representation

We design the *thread* visualization for representing multiple link variables with a focus on the relationship in an origin-destination pair (R1). Each network link consists of multiple *threads*, and each *thread*'s width is scaled based on a link variable's value. Therefore, each link has the same number of *threads* as the number of link variables, but with varying *thread* widths. While GreenGrid [41] utilizes the force-directed layout [28] and presents a (combined) variable on its links, the *threads* are placed on physical locations and present multiple variables. Users can choose the node variables to be encoded in the *thread* link width. Fig. 2 illustrates an example presenting how link variables can be mapped to *threads*. In this work, we use the departure delay times for each cause of delay as the link variables. This visual representation helps users easily identify which link has the largest delay and which delay type contributes most to the delay. In addition, when a link is specified as an anomalous link, it is located on the top in the stack of *threads* and other links become transparent so that the anomalous link can be highlighted as shown in Fig. 1 (F). Note that Bezier curves are utilized for the link visualizations, and *threads* can be sorted (e.g., departure delays or anomaly scores in our implementation). To help user perception, our system provides zooming (with a mouse wheel) and allows users to select the *thread* base width.

3.1.2 Petal Visualization

We introduce *petals*, a new directionally-aggregated radial visual representation as shown in Fig. 3 (Dallas, TX). In this representation, we can provide aggregated directional multivariate network link visualization with minimal visual clutter because we avoid link crossings [7]. Moreover, the spatial and multivariate characteristics are preserved and emphasized. Each directional *petal* (DP) encodes various information between one origin and multiple destinations in a given aggregate direction. Many transportation and logistics problems do have variable variation that is directionally dependent due to transportation paths, weather, routing, etc. By radiating from the origin location to multiple directions (one- to-many), a *petal* presents the geospatial relationships (R2). The *petal* length encodes a selected variable value (R2). Additional variable information is then encoded as radial sections within each *petal* (R3). For example, with the flight delay network data, the average departure delay for the flights heading for airports in a certain radial direction is mapped to the length of the *petal*. Then, the five types of delays are encoded inside the *petal* presenting the contributions of each delay type. Thus, we interpret that DP2 in Fig. 3, has a large NAS (National Aviation System, pointed by an red arrow) delay from Dallas. This indicates a large air traffic delay for the destinations, especially toward the airports in New York. Within a *petal*, we insert a pie chart visualization to show comprehensive overviews and comparisons among multiple variables in a node (R4). In the system, users can turn the *petal* display on and off. By default, we assign 12 *petals* for each origin but the users can merge two adjacent *petals* or split one *petal* into many *petals*. To help users easily recognize the destinations included in a *petal*, our system provides *petal* coverage guide lines as shown in Fig. 3. In addition, when the mouse hovers on a *petal*, the destinations included in the *petal* turn red for better recog-

nition. Lastly to ease comparison of *petal* lengths, equal-radius circles are drawn on all *petals*. The radius of the circle is the length of the *petal* where a user's mouse is hovering (e.g., the radius of the current circle in Fig. 3 is the length of DP2). Note that the data for the destinations within a *petal*'s coverage are aggregated and visualized together in the corresponding *petal*.

3.2 Pixel-oriented Network Matrix

Pixel-based visualization [20] is a technique for visualizing multivariate data but it can also be used for presenting cross-variable data (e.g., origin-destination pair data) by placing pixels side-by-side. We utilize this flexibility in our system to provide more complete multivariate network information, as shown in Fig. 1 *(G)*. Our system allows up to three square pixel matrices, where the *y*-axis of all matrices are the origins while the *x*-axis represent destinations. For example, for a flight delay network data for 20 airports, we place the departure delay matrix in *(G-1)*, the dominant delay type matrix (e.g., weather, security) in *(G-2)*, and the anomaly Z-score matrix *(G-3)* from our information-theoretic model as discussed in Section 4. Note that a Z-score filter is applied so that red pixels have Z-scores larger than 2 (97.7%) and purple pixels have Z-scores between 1 and 2 (84.1%). In our implementation, users can optionally make *G-3* present additional delay information (e.g., delay by airplane ages and by airlines as shown in Fig. 7 (c) and (d)). When a mouse hovers on a pixel, a tooltip pops up to display detailed information including delays of different types, the number of flights, and the anomaly scores, as shown in Fig. 1 (G-1). This interaction method is useful when a user wants to find out whether a delay type presented as a dominant type in (G-2) is indeed dominant among all delay types.

3.3 Time Series Displays

In order to present temporal trends, our system provides various time-series views: a calendar view *(A)*, clock view *(B)*, and line graph view *(C)* in Fig. 1. With the calendar representation [40] that applies a calendar metaphor to effectively reveal seasonality and cyclic trends, our system presents the delays by using different shading levels. For instance, the longer delays are presented with darker red. In addition, to help users identify any holiday effect, the week including a holiday has a yellow background. In order to supplement the functionality of the monthly trend line graph, our calendar representation provides additional weekly information on the right side of the calendar *(A-1)* and day of weekly patterns at the bottom of the calendar *(A-2)* in Fig. 1. The clock representation *(B)* is an efficient tool to detect daily trends [15], and we encode variables using areas to enhance visual perception according to Stevens' power law [35]. The line graph view *(C)* presents the types of aggregated delays as well as statistics such as the number of total flights, delayed flights, and average delay time.

3.4 Twitter Information Display

Analysis of Twitter messages (Tweets) generated within operational locations increases situational awareness and provides further insights for investigations by understanding responses from people and analyzing relationships between the responses and the data [11, 31, 36]. In this case, Twitter messages can be a good reference. In order to enable such analysis, we have set up a tweet collection system where our framework can retrieve tweets (posted after November 2011). In the collecting process, the text content of the Tweets is tokenized into words that are stemmed before the queries. Based on this infrastructure, our system provides Twitter view as shown in Fig. 1 (L) that can assist a user in examining the responses from people which were triggered by their delayed flights, and in finding additional information and correlation from the extracted related key tags. Once the user clicks one of the tags in the view, the dates of Tweets containing the tag are highlighted with blue outlines in the calendar view as shown in Fig. 6. We use opacity to encode the word frequency. Moreover, if the user selects a day in the calendar view, the actual Tweets of the day are displayed in the Tweet tab for further analysis.

4 Anomaly Detection and Highlighting

The visualizations in our system are able to draw upon an information-theoretic model for anomaly detection in a context-sensitive manner, utilizing the anomaly data for a consistent highlighting strategy shown throughout the visualization pipeline. For example, while Fig. 1 *(G-3)* explicitly encodes the anomaly score as the primary visual attribute, Fig. 1 *(F)* focuses on highly anomalous routes with thin outlines. In this case, attribute $a_{origin} = DFW$ (Dallas) is set as the condition in the model. What defines an 'anomalous' record depends upon the user's design and definition of individual anomaly detectors, *QCATs*, discussed in detail in this section. From a visual analytical perspective, these QCATs provide an overview of records where important attributes deviate from usual for specific conditions.

4.1 Overview of Anomaly Detection Method

Chandola et al. provided a comprehensive survey on methods for anomaly detection [12], categorizing them based on the nature of inputs, instance types, algorithmic mechanisms, and forms of outputs. For multivariate network data, we are interested in methods that can:

- Handle multi-dimensional records – because the main flight data concerned is a structured data stream consisting of 29 attribute dimensions (e.g., ≥ 10);
- Address the need for detecting contextual anomalies – which can provide a high-degree of flexibility and accommodating dynamic data and task variations in different detection scenarios;
- Facilitate an unsupervised algorithmic mechanism – alleviating the lack of training data in many situations;
- Generate anomaly scores as outputs that can be effectively conveyed by most visualization techniques.

In general, the family of statistical and information-theoretic methods can address the above-mentioned requirements better than the families of classification-, nearest neighbor- and clustering-based methods. As information theory is fundamentally built on probabilistic and statistical measures, information-theoretic methods may also be considered as a subset of the family of statistical methods. In this work, we use an information-theoretic method because of advantages as highlighted in [12]. "(1) They can operate in an unsupervised setting. (2) They do not make any assumptions about the underlying statistical distribution for the data."

Let $\mathbf{A} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n\}$ be a set of n variables. Each data record, $R = \{v_1, v_2, \ldots, v_n\}$ be a n-tuple, where v_i represents a valid value of attribute $\mathbf{a}_i$. In a practical scenario, an attribute, $\mathbf{a}_i$, may have a very large or infinite number of valid values. Binning is normally used to facilitate more accurate estimation of the probability of each valid value. In the following discussion, the probability distribution of an attribute, $p(\mathbf{a}_i)$, is assumed to be estimated in conjunction with an appropriate binning scheme.

The attribute set, $\mathbf{A}$, is divided into three mutually-exclusive subsets, $\mathbf{A}_{cnd}$, $\mathbf{A}_{von}$, and $\mathbf{A}_{ins}$. As anomalies are context-sensitive, $\mathbf{A}_{cnd}$ defines the context of a type of anomaly as a particular condition, such that all attributes in $\mathbf{A}_{cnd}$ are associated with specific values. For example, we may have $\mathbf{a}_4 = 1(Monday)$, $\mathbf{a}_{17} = JFK$, $\mathbf{a}_{18} = LHR$. The attributes in $\mathbf{A}_{cnd}$ are referred to as *conditional attributes*. In some situations, a conditional attribute may also take a range of values, e.g., $\mathbf{a}_4 = 1, 2, 3, 4$ or 5 (Monday–Friday).

The attributes in $\mathbf{A}_{von}$ play the primary role in determining an anomaly score for each record that has met the condition defined by $\mathbf{A}_{cnd}$. These attributes are referred to as *Variants of Normality* (VON). The remaining attributes, which are grouped into $\mathbf{A}_{ins}$, are considered to have "insignificant" influence on the type of anomaly concerned and are therefore excluded in the computation. Such a decision is usually made based on some known factors or logical reasoning by the user.

A combined configuration of $\mathbf{A}_{cnd}$ and $\mathbf{A}_{von}$ in relation to the overall attribute set $\mathbf{A}$, subsequently, determines how anomaly scores are estimated for each record. Given a record R, we first retrieve all records

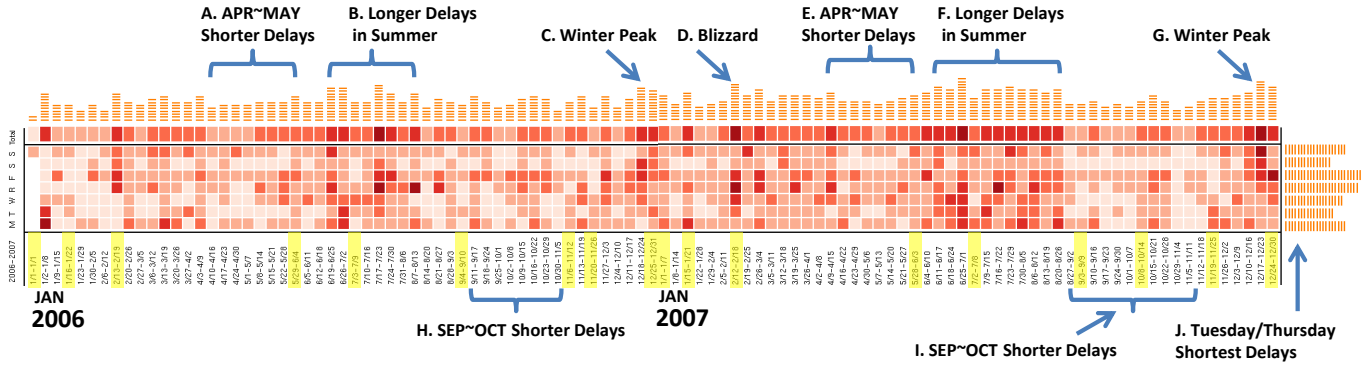


Fig. 4. Calendar view showing delay patterns for 2006–2007. In general, there were long delays in the summer and winter seasons, while APR–MAY and SEP–OCT did not have as many delays. Some delays increased around the holidays, but not all holidays had much impact on the delays.

that have the same conditional attribute values as R. Let this collection of records be $R_1, R_2, \ldots, R_W$, where W is usually a very large number. We now consider only the variants of normality defined by $\mathbf{A}_{von} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_j, \ldots, \mathbf{x}_s\}$. In conjunction with a binning scheme, each attribute, $\mathbf{x}_j$, may take valid values that are mapped to a set of t_j bins $B_j = \{b_{j,1}, b_{j,2}, \ldots, b_{j,t_j}\}$. For the s attributes in $\mathbf{A}_{von}$, there are a total of: $t_1 \times t_2 \times \ldots \times t_s$ different combinations of bins across different attributes. These combinations collectively define an alphabet $\mathscr{Z}$, and each unique combination is a letter $z \in \mathscr{Z}$.

The selection of an appropriate binning scheme for each attribute $\mathbf{x}_j$ is essential for ensuring that the total number of letters $|\mathscr{Z}|$ is smaller than the total number of records W. Ideally, we have $|\mathscr{Z}| << W$. We can, then, estimate the probability of each letter $z \in \mathscr{Z}$ based on the collection of records $R_1, R_2, \ldots, R_W$, resulting in a probability distribution function $p(z)$. For the given record R, we obtain its probability $p(R)$ by mapping it to its corresponding letter in $\mathscr{Z}$. The level of self-information is $I(R) = -\log_2(p(R))$, which is also called *surprisal*. We use this surprisal value as the anomaly score for the given record R. The level of uncertainty of this score can be defined as $H(\mathscr{Z})/\log_2(|\mathscr{Z}|)$, where $H(\mathscr{Z})$ is the entropy of the alphabet $\mathscr{Z}$.

It is necessary to emphasize that the anomaly score obtained for R reflects only the type of anomalies encoded by the specific configuration of $\mathbf{A}_{cnd}$ and $\mathbf{A}_{von}$. Hence, each configuration is only for queries of a specific type of anomaly in a particular context. We call each configuration a QCAT (**Q**uery **C**onditional **AT**tributes). It is not difficult to see that a visual analytics system can be equipped with one or more QCATs. For a given record, scores obtained using different QCATs can be aggregated, though it is necessary to understand the semantic implication of combining different QCATs and the difference between different aggregation methods (e.g., mean or max). Section 5.2 discusses the workflow for working with QCATs in a visual analytical system.

4.2 Implementation & Scalability

We have conducted a series of tests on the scalability of QCATs. Two implementations, client- and server-based, have been developed using Postgres. The former performs the grouping and aggregation on the client (i.e, in native code), and the latter uses a stored procedure hosted by the database server. Both server- and client-based implementations show that QCATs are linearly scalable in relation to the number of records used in the computation; the server-based implementation is about 2.5 times faster than the client-based implementation. Additionally, the client implementation is more sensitive to the network bandwidth and latency to the database server.

In our scalability tests, we have found that the performance of the server-based solution can be seriously affected by the number of VONs in $\mathbf{A}_{von}$, while the client-based implementation shows steady linear scalability in relation to the increasing number of VONs. The largest factor is the amount of shared buffers provided to Postgres. The scalability of entropy computation is linear but does rely on recomputing past data due to updated probability masses. However, Arackaparambil et al. [6] show that a distributed method for conditional entropy computation is feasible, while Guba et al. [16] demonstrate entropy estimation in streaming insert-only datasets. In the following sections, we describe how our system presents multivariate network data and visualizes the detected anomalies.

5 Geospatial Multivariate Network Data Exploration

As an example, we will use flight delay data from the Bureau of Transportation Statistics (BTS) [1] where each data row provides information for an individual flight including origin, destination, day of week, day of month, scheduled (departure/arrival) time, and real (departure/arrival) time and type of delay. There are five types of delays. Carrier delay is a problem within the airlines' control including mechanical problems of aircrafts, while NAS delay is caused by the control of the National Aviation System (NAS) including heavy traffic volume. Late Arrival Delay (LAD) is caused by the late arrival of the same aircraft at a previous airport. Security delay includes re-boarding time due to security breach and waiting time at the screening equipment. Weather delay means delay caused by extreme weather conditions at point of departure or arrival. Note that NAS delay and Security delay might be caused by the government organizations, while Carrier delay and LAD are caused by the airlines. We use the top 50 airports according to the number of passenger boardings that encompasses FAA's OEP-35 (Operational Evolution Partnership 35) airports accounting for more than 70% of the entire number of passengers [2].

5.1 Flight Delay Network Exploration

In this section, we explore the flight delay network data from 2006-2007 and summarize delay patterns in terms of temporal (e.g., summer, winter, holidays, weekly, hourly, and day of week) and spatial effects including special conditions such as severe weather (e.g., blizzards). First, we use the calendar view to investigate data patterns. In Fig. 4, we can see long delays as prominent seasonal patterns in the summer (B, F) and winter (C, G), while shorter delays were recorded during April–May and September–October. Another visible pattern is that there were fewer delays on Tuesday and Saturday in (J). We find that the patterns are related to holidays that are concentrated in summer and winter (e.g., Independence day in July, Christmas in December, personal vacations) but long delay patterns are not indicated for Martin Luther King day in January and Labor day in September. Moreover, long delay patterns tend to increase in 2007, especially in the summer (B and F). Also, there is a sudden spike (D) shown with the darkest red that might be another point for investigation.

Next, we can explore the aggregated delays for two years in the pixel view as shown in Fig. 5 (a, b), where we see some interesting patterns. The most prominent pattern is the series of horizontal and vertical dark red pixels (long delays) generated at the Chicago O'Hare

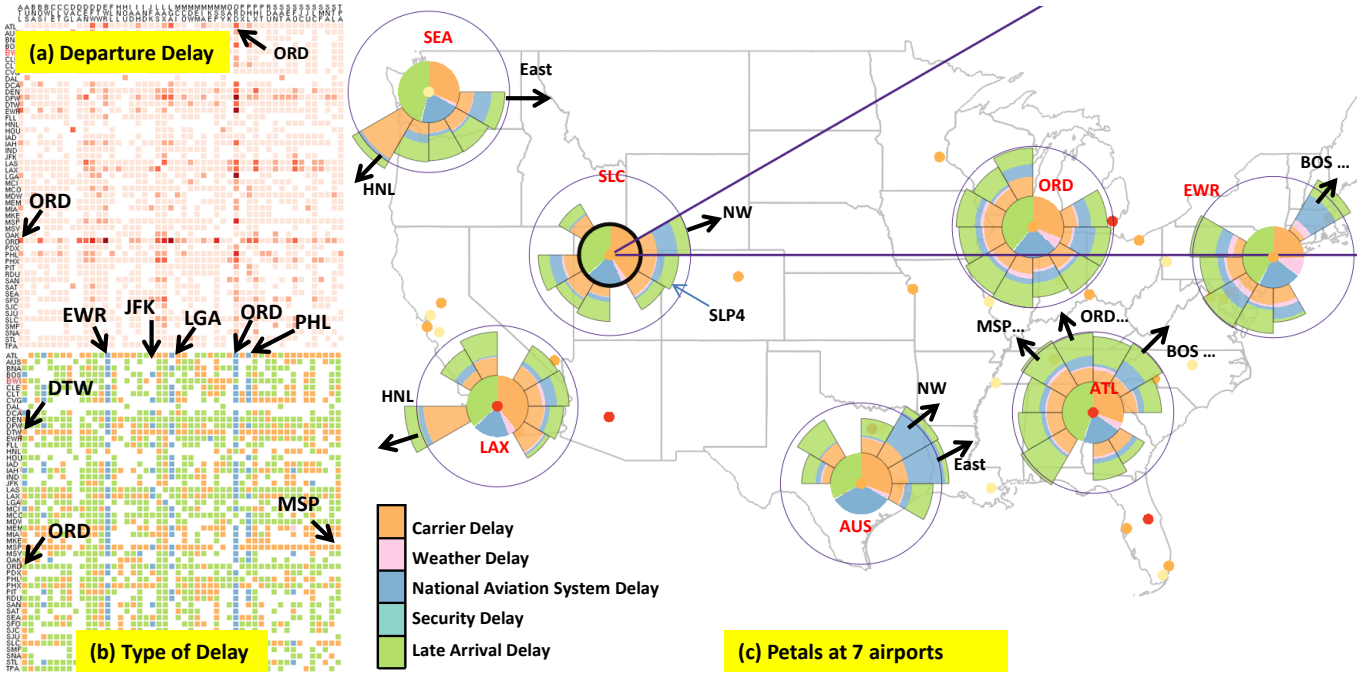


Fig. 5. (a) ORD is the most congested airport for both in-bound (vertical) and out-bound (horizontal). It is notable that carrier delay is the prevalent (out-bound) delay for DTW and MSP while NAS delay is the prominent delay for the incoming flights (vertical) in at EWR, JFK and LGA (b). (c) Flights heading to Hawaii from west cost airports in winter had long delays. Flights heading for ORD, ATL and airports from mid-east and east usually suffer from NAS delays.

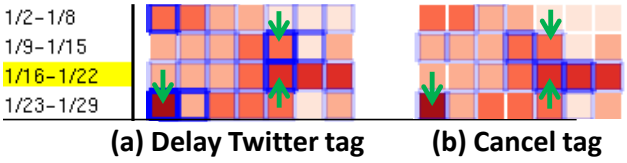


Fig. 6. Possible correlation between Twitter messages and delays. The opaque blue lines represent the day when many users posted messages related to the delay. The tag "delay" is selected in (a) and the tag "cancel" is selected in (b). Those days with arrows had severe weather reports on January 2012.

Airport (ORD) in (a), which indicates that both in-bound (horizontal) and out-bound (vertical) flights were severely congested. We also observe that such delays in ORD were caused mainly by late arrivals of aircraft (green) shown in (b). In addition, we notice that there are five distinguishable vertical blue lines in the matrix (b) and four of them (EWR, JFK, LGA, and ORD) were regulated by the High Density Rule (HDR) enacted in 1969 by the FAA due to severe congestion. This may indicate that the rule might not be strong enough to prevent such long delays. The delays in DTW (Detroit) and MSP (Minneapolis), which are two of the biggest hubs of Delta Airlines, are not very long compared to those in other top congested airports. However, it is interesting that the major type of delay is carrier delay (orange) caused by the airline itself.

Since one of the highest delays is observed in winter as shown in Fig. 4, we use our *petal* visualization with winter seasonal data for finding patterns and types of delays in the network as shown in Fig. 5 (c). We can select as many *petals* as designed for the exploration as long as minimal visual clutter is maintained. One interesting finding is that the flights heading for HNL (Hawaii) from the west coast airports (SEA and LAX) have relatively long delays (e.g., 120 minutes on average) and the prevalent cause for the delay is the carrier delay. Moreover, those airports also have relatively long NAS delays for flights heading for north-east destinations (ORD, and airports around New York).

The next interesting aspect is the delay distribution by time as shown in Fig. 1 (B) in the proportional mode with area encoding for each delay type. Here, we see a trend showing that delays increased from 6 am and had a peak around 6 pm. It is noted that this is the same pattern shown in the late aircraft delay while other types retained their proportion. This suggests that delays propagate during the day, a problem that Mazzeo termed "cascading delays" [26]. Such trends may imply that delays might be effectively reduced because these delays can be controlled either by the airlines (carrier delay/late arrival delay) with enough of an interval or layover time between two consecutive flight schedules, or by a government agency (e.g., Federal Aviation Administration) with advanced systems for air traffic control.

During exploration, we found that ORD (Chicago) generally shows longer delays than others in winter. To find any correlation between delay and Tweets, we use the twitter view with Tweet data generated within 3 miles of ORD in January 2012, as shown in Fig. 1 (L). We can see that several related Twitter tags occur, including as Chicago, airport, flight, delay, cancel, late, and hour. When delay and cancel tags are clicked, we find a possible correlation between the tags and the delays on the calendar (January, 2012) as shown in Fig. 6. In the figure, the opaque blue outlines presenting frequencies of Tweet messages are placed on the dates when large delays were recorded.

Of primary interest are the patterns in the length and types of delays that can be can be better explored by sorting airports. We see that the ranks change with little variation based on seasons, but most delays are caused by some major airports including ORD (Chicago), ATL (Atlanta), LGA (New York City), EWR (New York City), DTW (Detroit), LAX (Los Angeles), LAS (Las Vegas), and DFW (Dallas) as shown in Fig. 7 (a). From the type matrix Fig. 7 (b), we notice that in many highly-ranked airports, the main type of delay is the late arrival delay in busy travel seasons while the NAS delay is dominant at other times. This implies that the NAS might not be properly adapting to the current increasing traffic in terms of delays. On the other hand, we notice that the two distinguishable airlines causing delays are AA (American Airline) and UA (United Airline) in the two most delayed airports as shown in Fig. 7 (c). The dominant delay type matrix in Fig. 7 (b) indicates that the airlines are responsible for solving

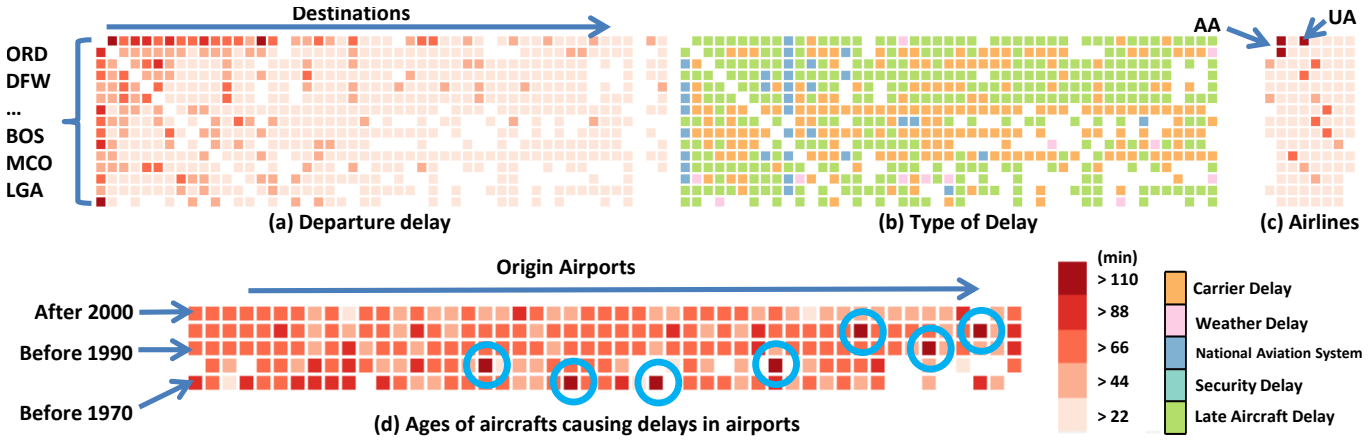


Fig. 7. Airports are sorted by delays. ORD shows the longest delays in many out-bound flights in (a). The dominant type of delay was carrier delay and LAD in (b). UA and AA had the longest delays in ORD when ORD was top by delays in (c). Old airplanes generally caused long delays in (d).

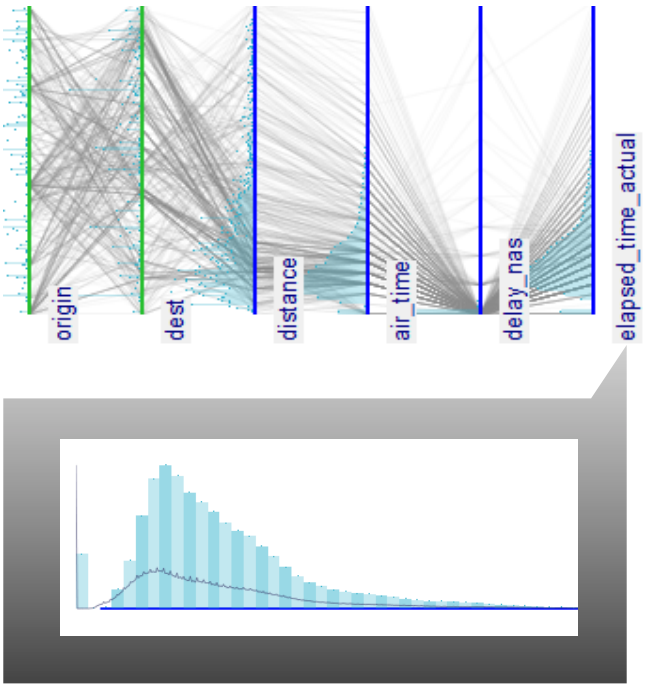
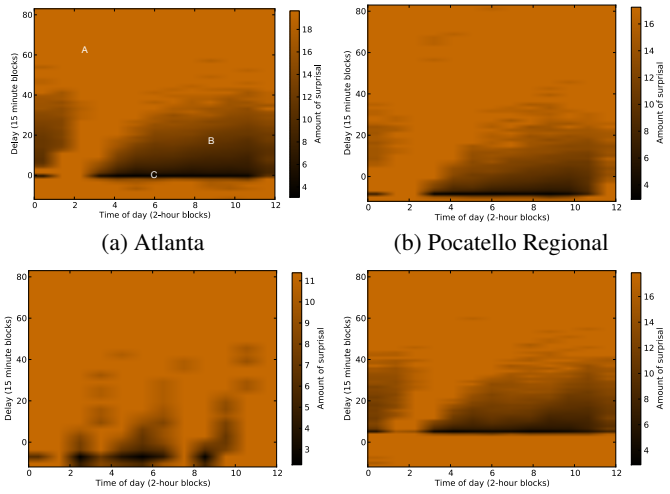


Fig. 8. Using Parallel Coordinates to Design QCATs: *(top)* Exploring the attributes as a parallel coordinate plot; *(bottom)* Specifying an individual attribute's bin specification



(a) Atlanta (b) Pocatello Regional

(c) Sitka (d) John F Kennedy Intl

Fig. 9. Heatmaps representing the surprisal spaces of flights leaving four different airports, with *(x)* Time of day *(bin size: 2 hrs)*, and *(b)* Departure delay *(bin size: 15 mins)*

the delay problem because the dominant types of delays were carrier delay and late aircraft arrivals. Finally, it is also noted that there are old airplanes operating as shown in (d, rotated) where we see that 7 airports have the darkest red colors and only the older airplanes cause the severe delays. It is also notable that airplanes manufactured before 1970 have higher average delays across airports.

5.2 QCAT Workflow

As discussed in Section 4, our system features an information-theoretic anomaly detection system that is comprised of a set of user-defined QCATs. The design of a QCAT can be based on a specific hypothesis, or as a more general monitoring system for one or more attributes. Ideally, in a deployed system, the roles of QCAT *designer* and overall *analyst* would be disparate, with the analyst analyzing the data for anomalies and reporting back to the designer to refine the QCATs based on new trends.

To assist the user in defining the QCATs in the system, we provide a designer tool based on parallel coordinates (see Fig. 8 *(top)*) while the user is able to explore the attribute space by adding/removing attribute dimensions, observing their value distributions (e.g., probability mass functions), as well as viewing the record relationship between attributes afforded by a standard parallel coordinates representation. The role of an attribute can be toggled between conditional (green) and VON (black) using the right mouse button. The user is also able to explore an individual attribute in more detail by clicking the left mouse button that expands the attribute to the full view to show its distribution in more detail (Fig. 8 *(bottom)*). The detail view also shows the attribute's bin width specification, which can be modified per QCAT. The user's choice of bin width has an effect on the anomaly results and reflects the user's knowledge of the attribute's semantic meaning. The system maps data types to suitable bin width granularities automatically. For example, timestamp datatypes are divided into bins of n minutes; categorical data such as strings are unbinned. Since integer types may represent categorical, interval or ratio measurements, we assume a default bin width of 1 and let the user decide upon a more suitable width.

Once the user has defined a QCAT, it can be saved to the QCAT library and selected as the active QCAT. Anomaly-supporting visualiza-

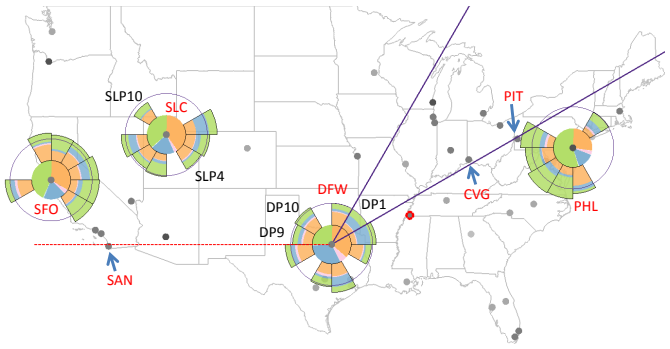


Fig. 10. An example for a *petal* experiment. With the visual aid, users could better tell that CVG is included in DP1 while PIT is included in DP2.

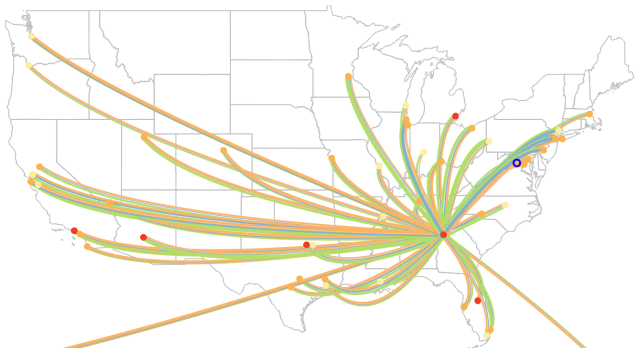


Fig. 11. An example of a *thread* experiment. 40% of the participants answered incorrectly that green was the prevalent delay due to the severe color concentration around the origin.

Petal Index	Difference (%)	Accuracy (%)	Time (s)
DP1	4 (Small)	76.7	8.2
DP10	21 (Large)	100	3.7

Table 1. Participants found the longest delay inside a *petal* more accurately in less time as the difference became larger (HP1).

Petal Index	Difference (%)	Accuracy (%)	Time (s)
DP9	3 (Small)	46.7 (83.3)	6.6 (5.4)
SLP10	12 (Large)	96.7 (100)	3.9 (2.6)

Table 2. As the difference became larger, the participants better detected the shortest delay (HP2). Visual aids improved both the accuracy and efficiency (HP5).

tions in our system such as the pixel-oriented network matrix update to reflect the anomaly scores by completing the relevant conditionals in the QCAT (i.e., origin and destination pairs) and executing the QCAT on the data to obtain statistics (i.e., mean, max, variance) on the surprisal values for records matching the conditionals. Our system by default displays the maximum surprisal value as the anomaly value mapped to a visual attribute (i.e, halo) in the visualization. The anomaly values in the visualizations guide the user to identify abnormal flights based on their own criteria specified in the design of each QCAT. Anomalous results can then be explored further using the available visual analytical tools to understand why the anomaly value was high and report these findings to the QCAT's designer.

For a QCAT consisting of two VONs, we can illustrate the anomaly distribution using a heatmap. Fig. 9 shows the anomaly space for flights leaving four different airports for the years 2006 and 2007. The x-axis shows the time of day divided into two-hour blocks, and the y-axis shows the amount of delay in 15-minute blocks (notice that flights can leave early). Areas of low surprisal value are black and become amber with higher surprisal values. It is clear that for this airport, flights around 4AM are uncommon, and the amount of delay seems to increase steadily throughout the day until late afternoon before leveling out. For the Atlanta airport, three example records, A, B and C are shown of high ($\approx$19.68), slightly above average ($\approx$14.36), and low ($\approx$3.478) surprisal values, respectively. Investigating these flights using *threads* shows that late aircraft were largely to blame for both A and B; however, in the case of A the high surprisal value indicates that such a large delay is unusual at this time of the morning. At C, we find ourselves in the 'usual' low-anomaly area for this airport, where delays are close to zero for most of the day.

6 User Study

In order to evaluate the *petal* and *thread* designs, we performed a user study with 30 participants recruited from various majors at our university. In the study, the participants were given computer-based tasks for verifying hypotheses. Various difference levels in the flight delay network data were used in the tasks. Note that the *difference level* in this section means the difference between the longest (shortest) and second longest (shortest) delays. Note that the numbers in parentheses in the summary tables are the results with visual aids. We use a paired t-test to check if our experimental result obtained is significant (p-value $<$ 0.05) within a 95% confidence interval.

6.1 Petal User Study Results

We first set up the following hypotheses for the *petals* visualization as follows:

HP1 As the difference becomes larger, users will show high accuracy and speed in detecting the longest delay inside a *petal*,

HP2 As the difference becomes larger, users will show high accuracy and speed in finding the shortest (or longest) delay among *petals* for one operational place,

HP3 Users will show lower accuracy in finding the shortest (or longest) *petal* among the *petals* at multiple operational places,

HP4 Users will show low accuracy and speed in finding whether an airport is included in *a petal* as the distance between the *petal* and the airport becomes longer and as an airport is close to the boundary of the *petal*, and

HP5 Visual aids will improve accuracy and speed.

TASK1 for verifying HP1 asked the participants to choose the longest delay inside a *petal* in 2 locations: DP1 (delay difference: 4%)and DP10 (21%), as shown in Fig. 10. The participants showed higher accuracy and speed as the difference increased (Table 1, p-value $<$ 0.05). In TASK2, for verifying HP2, the participants were asked to select the shortest *petal* in 2 locations: DFW (3%) and SLC (12%). For a small difference (3%), 46.7% of the participants answered correctly. As the difference became larger and the visual aid (circle) was provided (HP5), both accuracy and speed were improved (Table 2, p-value $<$ 0.05). TASK3 was the same as TASK2 but multiple *petals* at Salt Lake City (SLC), San Francisco (SFO), and Philadelphia (PHL) were presented concurrently. Here, the participants showed lower accuracy (from 46.7% to 36.7%) and slower speed (from 6.6s to 10.9s) compared to the results in TASK2. The visual aid (HP5) improved both accuracy and speed in the lower difference (Table 3, p-value $<$ 0.05). In order to evaluate if users accurately recognized the coverage of each *petal* (HP4), TASK4 asked the participants to select airports that were included in DP1 and DP9, as shown in Fig. 10. As summarized in Table 4, the participants showed low accuracy (23.3% and 60%). The main reason for such low accuracy was that it was hard for them to find whether CVG (Cincinnati) and PIT (Pittsburgh) were included in DP1. In the same context, only 60% of the participants

Petals	Diff. (%)	Accuracy (%)	Time (s)
SLC+SFO+PHL	2 (Small)	36.7 (73.3)	10.9 (6.8)
SLC+SFO+PHL	12 (Large)	96.7 (96.7)	4.7 (5.1)

Table 3. Users had difficulty finding the longest delay among distant *petals* with a small (2%) difference (HP3). The visual aid helped the users better answer with a small difference (HP5).

Petal Index	# of Airports	Accuracy (%)	Time (avg.)
DP1	8	23.3 (83.3)	1.96 (1.18)
DP9	8	60.0 (93.3)	2.3 (1.3)

Table 4. The participants had difficulty finding whether CVG and PIT were included in DP1 (HP4). The visual aid helped the users better recognize if an airport was included in a *petal* or not (HP5).

correctly found that SAN was not included in DP9. However, with the visual coverage line (HP5), both the accuracy and speed improved.

6.2 Thread User Study Results

Next, we set up hypotheses for the *threads*. As the difference between the longest and the second longest delays becomes larger, the users will produce better results in HT1) detecting the longest delay inside the *threads*, and in HT2) choosing the most prevalent delay among all the *threads*. TASK5 for verifying HT1 asked the participants to select the color of the thickest *thread* for small (3.1%) and large (28%) difference levels. As summarized in Table 5, when the difference was small (3.1%), it was hard for the participants to tell the longest delay (66.7% accuracy). On the other hand, when the difference became larger, they answered very accurately and spent less time (p-value $<$ 0.05). TASK6 for verifying HT2 asked the participants to tell the color of the longest delay when all the *threads* were considered. Here we see a similar result as in TASK5: the larger the difference, the higher the accuracy and the slower the speed (Table 6). In TASK6, we had an interesting result showing that special concentration of a color may interfere with accurate visual perception. For example, we can see LAD (green) is concentrated on short-haul routes as shown in Fig. 11. In this case, 40% of the participants thought that LAD was the longest delay for flights leaving from Atlanta, but in fact the carrier delay was 23% larger than LAD. This error rate is unexpected compared to the result in TASK5 where the participants showed higher accuracy and speed with a similar difference (28.6%). Conversely, we think it is possible that users could assume that the color on long-haul routes has the largest value if the color is concentrated in long-haul *threads*. To prevent this, our system provides numeric information in the legend view that users can refer to, as shown in Fig. 1 (E).

7 LIMITATIONS AND DISCUSSION

Petals have a similar appearance to the rose diagram (or sunburst) visual representation that has been adapted in various contexts [14, 27, 34]. The contribution of *petals* lies in extending the usability of the family of the rose diagram by allowing geographically-directional, multi-variate, and aggregated network analysis simultaneously. Discerning widths of *thread* can be hard when each variable has similar values or when a unit *thread* within a route is not thick enough for visual perception. In addition, when a color is concentrated on long-haul or short-haul routes, it could be hard to select the largest value among all *threads*. To help users with those issues with threads, our system provides the numeric information of the variables in the legend view when a user specifies an area of *threads* (aggregated) and in a tooltip when the user's mouse hovers over an airport (origin to destination). The tooltip in the pixel view can be used for verifying that the presented dominant delay (Fig. 1 (G-2)) is indeed dominant compared to others.

Difference (%)	Accuracy (%)	Time (s)
3.1 (Small)	66.7	5.2
28 (Large)	100	2.9

Table 5. The participants made errors and spent more time finding the longest delay when the difference in *threads* was small (HT1).

Difference (%)	Accuracy (%)	Time (s)
11 (Small)	90	5.3
28.6 (Large)	100	2.9
23 (Large)	60	5.1

Table 6. 40% of the participants answered incorrectly with a large difference (23%) in finding the prevalent delay among all *threads*. This may indicate that color concentration on long-haul or short-haul *threads* interferes with visual perception.

8 CONCLUSION AND FUTURE WORK

We have explored complex multivariate network links with multiple tightly-integrated interactive visualizations. We have introduced two new visual representations, *petals* and *threads*, for spatial multivariate link visualization. Our sortable matrix displays have the ability to represent multiple origin and destination pairs with enhanced pixel-based visualization, while the linked line graph, calendar, and clock views give opportunities to find temporal characteristics. An information-theoretic anomaly detection model was introduced based on conditional attributes, with the visualizations in the system utilizing the surprisal values for visual highlighting of anomalies in multiple visualization components in a unified manner.

Our system has several benefits compared to previous systems. Our system allows users to investigate the data status of a large number of operational locations by simultaneously observing various data characteristics at both aggregate (entire network) and detailed levels (e.g., origin-destination pairs) using our multiple linked view. Our new visual representations, *petals* and *threads*, help users find features of multiple spatial network variables with minimum visual clutter; the pixel-based network matrices aid in analyzing the entire network in terms of multiple origin-destination pairs as well as origin-attribute pairs. Seasonal and cyclical trends can be efficiently detected in the calendar, line graph, and clock visualizations from our system. Lastly, our system provides an information-theoretic model for detecting anomalies based on conditions. For the evaluation of our system, we presented an example using flight delay network data from the top 50 airports to illustrate the use and potential of our designs and the user study results.

Our system can be easily applied to analysis with any other multivariate spatiotemporal, network-based data such as transportation and logistics, trading, and communication industries [7]. As a future work, we plan to incorporate the ability to help users find correlations using *petals* and *threads*. The capability for visualizing cascading effects and clusters of operational places that have the same characteristics will also be investigated. In addition, we would like to explore our anomaly detection more by investigating methods of combining the anomaly values for groups of QCATs.

REFERENCES

[1] Bureau of Transportation Statistics (Accessed 20 Mar 14. `http://www.rita.dot.gov/`.

[2] Operational Evolution Partnership 35. `http://aspmhelp.faa.gov/index.php/OEP_35`.

[3] Qlikview. `http://www.qlikview.com/`.

[4] Tableau. `http://www.tableausoftware.com`.

[5] C. Ahlberg. Spotfire: An information exploration environment. *ACM Special Interest Group on Management of Data Record*, 25(4):25–29, 1996.

[6] C. Arackaparambil, S. Bratus, J. Brody, and A. Shubina. Distributed monitoring of conditional entropy for anomaly detection in streams. In

Parallel Distributed Processing, Workshops and Phd Forum (IPDPSW), 2010 IEEE International Symposium on, pages 1–8, 2010.

[7] R. A. Becker, S. G. Eick, and A. R. Wilks. Visualizing network data. *IEEE Transaction on Visualization and Computer Graphics*, 1(1):16–21, Mar. 1995.

[8] R. Borgo, K. Proctor, M. Chen, H. Janicke, T. Murray, and I. Thornton. Evaluating the impact of task demands and block resolution on the effectiveness of pixel-based visualization. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):963–972, 2010.

[9] H. Bosch, D. Thom, M. Worner, S. Koch, E. Puttmann, D. Jackle, and T. Ertl. Scatterblogs: Geo-spatial document analysis. In *Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on*, pages 309 –310, 2011.

[10] D. Brooks. What data can't do. *The New York Times*, Feb. 2013.

[11] J. Chae, D. Thom, H. Bosch, Y. Jang, R. Maciejewski, D. Ebert, and T. Ertl. Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 143–152, 2012.

[12] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):15:1–15:58, July 2009.

[13] B. Duffy, J. Thiyagalingam, S. Walton, D. J. Smith, A. Trefethen, J. C. Kirkman-Brown, E. A. Gaffney, and M. Chen. Glyph-based video visualization for semen analysis. *IEEE Transactions on Visualization and Computer Graphics*, 99:1, 2013.

[14] N. Elmqvist, J. Stasko, and P. Tsigas. Datameadow: A visual canvas for analysis of large-scale multivariate data. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 187–194, 2007.

[15] J. Fuchs, F. Fischer, F. Mansmann, E. Bertini, and P. Isenberg. Evaluation of Alternative Glyph Designs for Time Series Data in a Small Multiple Setting. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI)*. ACM, 2013.

[16] S. Guha, A. Mcgregor, and S. Venkatasubramanian. Streaming and sublinear approximation of entropy and information distances. In *ACM-SIAM Symposium on Discrete Algorithms*, pages 733–742, 2006.

[17] D. Guo. Flow mapping and multivariate visualization of large spatial interaction data. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1041–1048, 2009.

[18] M. A. Harrower and C. A. Brewer. Colorbrewer.org: An online tool for selecting color schemes for maps. *Cartographic Journal*, 40(1):27–37, 2003.

[19] Herman, G. Melançon, and M. S. Marshall. Graph visualization and navigation in information visualization: A survey. In *IEEE Transactions on Visualization and Computer Graphics*, volume 6 (1), pages 24–43. 2000.

[20] D. A. Keim. Designing pixel-oriented visualization techniques: Theory and applications. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):59–78, 2000.

[21] D. A. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann. *Mastering The Information Age-Solving Problems with Visual Analytics*. Florian Mansmann, 2010.

[22] S. Ko, R. Maciejewski, Y. Jang, and D. S. Ebert. Marketanalyzer: An interactive visual analytics system for analyzing competitive advantage using point of sale data. *Computer Graphics Forum*, 31(3):1245–1254, 2012.

[23] Y. Kopylova, D. Buell, C.-T. Huang, and J. Janies. Mutual information applied to anomaly detection. *Communications and Networks, Journal of*, 10(1):89–97, 2008.

[24] W. Lee and D. Xiang. Information-theoretic measures for anomaly detection. In *Security and Privacy, 2001. S P 2001. Proceedings. 2001 IEEE Symposium on*, pages 130–143, 2001.

[25] Z. Liu, S. B. Navathe, and J. T. Stasko. Network-based visual analysis of tabular data. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 41–50, 2011.

[26] M. Mazzeo. Competition and service quality in the u.s. airline industry. *Review of Industrial Organization*, 22(4):275–296, June 2003.

[27] F. Nightingale. *Notes on Matters Affecting the Health, Efficiency, and Hospital Administration of the British Army*. Harrison and Sons, 1958.

[28] A. Noack. An energy model for visual graph clustering. In *Graph Drawing*, volume 2912 of *Lecture Notes in Computer Science*, pages 425–436. Springer, 2003.

[29] D. Oelke, H. Janetzko, S. Simon, K. Neuhaus, and D. A. Keim. Visual boosting in pixel-based visualizations. *Computer Graphics Forum*, 30(3):871–880, 2011.

[30] J. C. Roberts. State of the art: Coordinated & multiple views in exploratory visualization. In *Proceedings of Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization*, pages 61–71, 2007.

[31] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, pages 851–860, 2010.

[32] A. Z. Santovena. Big data : evolution, components, challenges and opportunities. Master's thesis, Massachusetts Institute of Technology, Sloan School of Management, 2013.

[33] R. Scheepens, H. van de Wetering, and J. J. van Wijk. Non-overlapping aggregated multivariate glyphs for moving objects. In *IEEE Symposium on Pacific Visualization*, pages 17–24, 2014.

[34] Z. Shen and K.-L. Ma. Mobivis: A visualization system for exploring mobile data. In *IEEE Symposium on Pacific Visualization*, pages 175–182, 2008.

[35] S. S. Stevens. *Psychophysics: Introduction to Its Perceptual, Neural, and Social Prospects*. Wiley, 1975.

[36] D. Thom, H. Bosch, S. Koch, M. Woerner, and T. Ertl. Spatiotemporal anomaly detection through visual analysis of geolocated twitter messages. In *IEEE Pacific Visualization Symposium (PacificVis)*, 2012.

[37] J. J. Thomas and K. A. Cook, editors. *Illuminating the Path: The R&D Agenda for Visual Analytics*. IEEE Press, 2005.

[38] F. van Ham, H.-J. Schulz, and J. M. DiMicco. Honeycomb: Visual analysis of large scale social networks. In *Proceedings of International Conference on Human-Computer Interaction*, volume 5727 of *Lecture Notes in Computer Science*, pages 429–442. Springer, 2009.

[39] Wattenberg, Martin. Visual exploration of multivariate graphs. In *Proceedings of ACM CHI 2006 Conference on Human Factors in Computing Systems*, volume 1 of *Visualization 1*, pages 811–819, 2006.

[40] J. V. Wijk and E. V. Selow. Cluster and calendar based visualization of time series data. In *1999 IEEE Symposium on Information Visualization (INFOVIS '99)*, pages 4–9, Oct. 1999.

[41] P. C. Wong, K. Schneider, P. Mackey, H. Foote, G. Chin, Jr., R. Guttromson, and J. Thomas. A novel visualization technique for electric power grid analytics. *IEEE Transactions on Visualization and Computer Graphics*, 15(3):410–423, May/June 2009.

[42] L. Zhang, A. Stoffel, M. Behrisch, S. Mittelstädt, T. Schreck, R. Pompl, S. Weber, H. Last, and D. A. Keim. Visual analytics for the big data era - A comparative review of state-of-the-art commercial systems. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 173–182. IEEE Computer Society, 2012.

[43] H. Ziegler, T. Nietzschmann, and D. A. Keim. Visual analytics on the financial market: Pixel-based analysis and comparison of long-term investments. In *Proceedings of International Conference on Information Visualisation*, pages 287–295. IEEE Computer Society, 2008.

AnnotatedTimeTree: Visualization and Annotation of News Text and Other Heterogeneous Document Collections

Jing Xia, Jieqiong Zhao, Isaac Sheeley, Joseph Christopher, Qiaoying Wang, Chen Guo, Jiawei Zhang, David S. Ebert, Yingjie Victor Chen, Zhenyu Cheryl Qian

Abstract—Investigation of critical events requires dealing with heterogeneous document collections, and is usually done case by case. We address the case of VAST Challenge 2014 Mini-Challenge 1 with the two views of our *AnnotatedTimeTree* system: *the document view* and *the GAStech view*. The document view is a novel AnnotatedTimeTree design for current and historical news reports in Kronos; and the GAStech view is implemented specifically for GAStech employee profiles and email activities. Together the two views facilitate rapid summarization of both crucial events in Kronos history and GAStech employees in Mini Challenge 1. The AnnotatedTimeTree design provides semantic interactions such as searching, highlighting, annotating and expanding for maintaining reasoning process and reasoning results to the documents.

Index Terms—Temporal visualization, document analysis

1 Introduction

Investigation of critical events requires putting together heterogeneous data collections. It requires human intelligence integration and thus is usually performed case by case. Good visual analytical tools can integrate human interactions and promote reasoning from existing data resources.

The VAST Challenge 2014 Mini-Challenge 1 is about investigation of missing employees in GAStech, a natural gas company. The suspect POK (protectors of Kronos) were originated from a small village, which had been long suffering from dirty water polluted by GAStech. Participants are given several datasets including GAStech employee profiles, resumes and email conversations, historical news reports regarding the conflicts between GAStech and POK. The tasks are to find out the development of POK, the event timeline, and to identify possible explanations why those employees were missing.

To address the problem of VAST Challenge 2014 Mini-Challenge 1, we developed AnnotatedTimeTree, a two-view online system to integrate the datasets: the document view for analysis of historical reports, and the GAStech view for exploration of GAStech employees. Specifically, the AnnotatedTimeTree in the document view is an innovative design we created to use a timeline to organize reports and analysts' annotations of these reports. It provides semantic interactions such as searching, highlighting, annotating and expanding for maintaining reasoning process and reasoning results to the documents. The GAStech view enables interactive investigations of employees based on given profiles, resumes and email conversations. Together with the two views, the AnnotatedTimeTree system promotes users reasoning process in discovering possible explanations of missing employees.

2 Visual Exploration

AnnotatedTimeTree is composed of a news report timeline, an annotation area, a navigation bar and a report reader. Interactions such as highlighting, navigating, filtering are provided for faster locating of crucial persons, locations etc. The GAStech view consists of an e-mail explorer, an employee resume viewer and an employee network visualization.

- *Jing Xia is with State Key Lab of CAD&CG, Zhejiang University in Hangzhou, China. E-mail: xiajing@zjucadcg.cn.*
- *Jieqiong Zhao, Isaac Sheeley, Joseph Christopher, Qiaoying Wang, Chen Guo, Jiawei Zhang, David S. Ebert, Yingjie Victor Chen, Zhenyu Cheryl Qian are with Purdue University in West Lafayette, IN, USA. E-mail: [illegible], zhao, isheeley, christ33, wang1925, guo171, zhan1486, ebertd, [illegible].*

IEEE Symposium on Visual Analytics Science and Technology 2014
November 9-14, Paris, France
978-1-4799-6227-3/14/$31.00 ©2014 IEEE

2.1 AnnotatedTimeTree Design

AnnotatedTimeTree (see Fig. 1) is a day-based report timeline with each block (the unit on the timeline) being green color coded to the number of reports concerning a certain person on a certain day. We picked 11 names that have the highest frequencies in the reports, with all the remaining persons falling into the "Other" category. Grey areas are applied to those days with no reports since the news report dataset given is not continuous. Because the overall timeline is too long to show, a zoomed-out overview is provided to the right for preview and navigation. To the left is the report pile area, where users can browse the reports one by one and make annotation. The annotations being made are aligned next to its timeline location for browse.

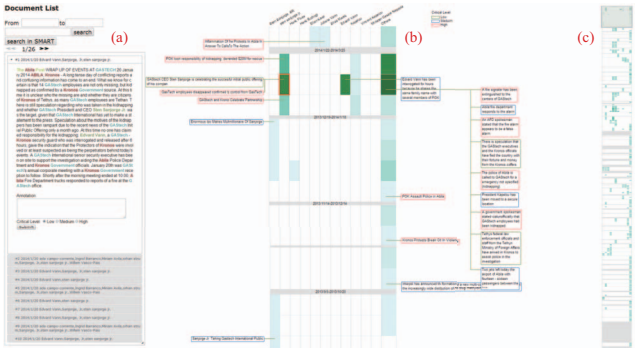


Fig. 1. The document view of AnnotatedTimeTree. From left to right there are the report pile area with search bar at its top (a), the timeline (b) with annotations attached to its left and right and the thumbnail navigation bar (c).

Navigate With the timeline preview, users are able to find out days of interest by locating dark green blocks. We also notice a pattern of two-column green blocks. After inspection of those blocks, it turns out to be all conflicts between POK and GAStech in memory of POK leader Elian Karel and the girl Juliana Vann. Users can also navigate to those days by clicking on the timeline preview.

Search If users want to narrow the reports down to a smaller amount, they may search on the left with duration of time and keywords. All results are highlighted with the given keywords.

Highlight Other than highlights of keywords, names of persons, locations and organizations are also highlighted with different colors

to draw users attention.

Annotate Users can make annotations or modify annotations at the end of reports. Based on the critical level of the annotations, they are color coded differently. Annotations to the same person of the same day are aligned and ordered in a string. Once changed, the corresponding annotation will be created or modified to maintain users' investigation on the report.

Expand When there are large numbers of reports accumulated in one day, users can expand the day by double clicking on a certain green block. Corresponding reports will be expanded and sorted according to time of day (see Fig. 2). Additionally, the annotations will be re-aligned correspondingly.

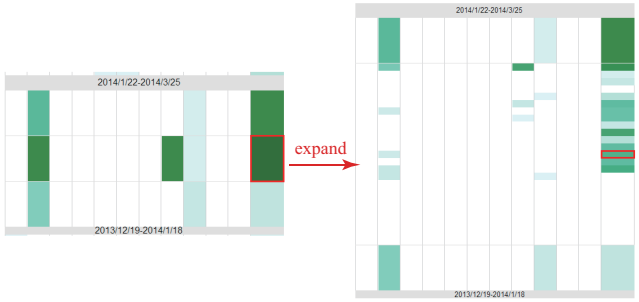


Fig. 2. Timeline of one day is expanded to 24 hours.

2.2 GAStech Profile Design

The GAStech view (see Fig. 3) integrates GAStech organization chart, employee resumes, profiles and their email conversations altogether in one view. A network based on the email conversation is built to explore employee relations. Employees are color coded according to the departments they belong. Users can mark emails as important and thus narrow the email network down to senders and receivers of important emails only. Users can also browse employees' profiles, resumes and car activities (for Mini-Challenge 2).

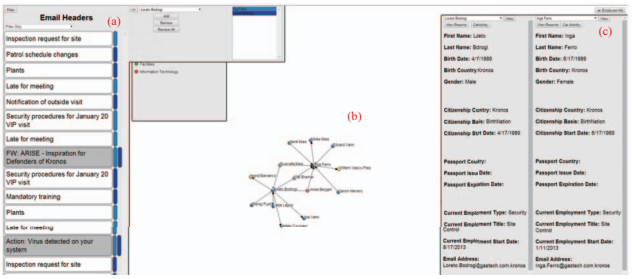


Fig. 3. The GAStech view of AnnotatedTimeTree. From left to right there are the email conversation viewer (a) where users can sort, mark and check details of the emails, the employee network (b) based on email conversations, and the employee resume viewer (c) where users can compare resume or profiles of any two employees.

2.3 Analysis Process

Fig. 4 shows that there had been several conflicts between POK and GAStech ever since the death of ten-year old girl Juliana Vann. The situation became worse after former POK leader Elian Karel's death. the annual rally was held and ended in riot.

By investigating the reports, users are able to efficiently locate those reports to summarize a timeline of what happened (see Fig. 1). First at 09:26 January 20th, GAStech CEO Sten Sanjorge was celebrating

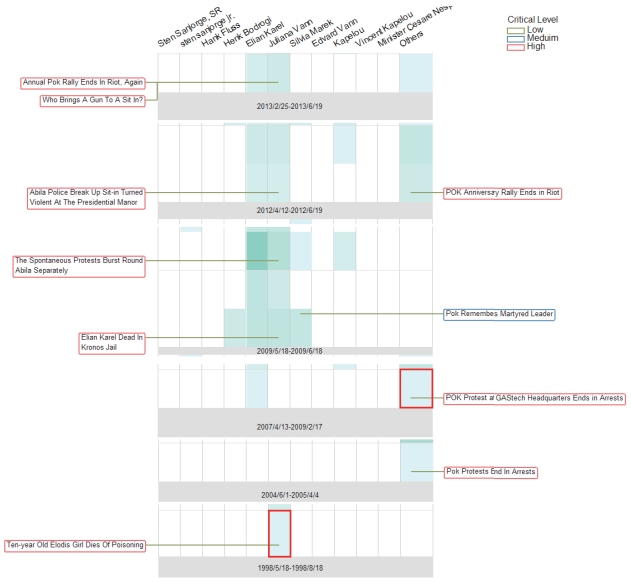


Fig. 4. Summary of historical conflicts between POK and GAStech.

the successful initial public offering of his company. Later the celebration was interrupted by a false fire alarm. Then at 12:20 the police of Abila were called to GAStech for an unspecified emergency. At 15:00 Edvard Vann was interrogated for hours. The last name "Vann" is shared by several key POK members. Though unofficially, a government spokesman stated that GAStech employees had been kidnapped at 20:20. At 22:00 Tethys federal law enforcement officials and staff from the Tethyn Ministry of Foreign Affairs arrived in Kronos to assist police in the investigation. On January 21st, 2014 POK announced the kidnapping and demanded 20M.

2.3.1 Findings

A reasonable explanation of employees' disappearance in GAStech might be they were kidnapped by GAStech employees who are POK members indeed. Two evidences support the finding. First, there had been a long history of conflicts (see Fig. 4) between POK and GAStech. Second, some GAStech employees from security department share the same family names with leaders or founders of POK, and they communicated frequently via emails (see Fig. 3). The responsible GAStech employees are Inga Ferro, Loreto Bodrogi, Isia Vann, Hennie Osvaldo, Minke Mies and Ruscella Mies, all from security department .

We also found something interesting (though not reflected in the figures), that different news media tends to have different opinions towards the same events. "Homeland Illumination" seemed to speak for POK while "The Abila Post" stood for governments' opinion.

3 Conclusion

AnnotatedTimeTree is an online system with two views for visualization of historical reports and GAStech profiles respectively. The AnnotatedTimeTree design in the document view allows users to make annotations on reports dynamically, which not only records and communicates users' understanding but also maintains their reasoning process towards the overall story. This design can be easily extended to analysis of time-critical documents.

AnnotatedTimeTree, Dodeca-Rings Map & SMART: A Geo-Temporal Analysis of Criminal Events

Chen Guo[1], Jing Xia[4], Jun Yu[1], Jieqiong Zhao[2], Jiawei Zhang[2], Qiaoying Wang[3],
Zhenyu Cheryl Qian[3], Yingjie Victor Chen[1], Chen Wang[5], David Ebert[2]

[1]Computer Graphics Technology, [2]Electrical and Computer Engineering, [3]Interaction Design, Purdue University

[4]State Key Lab of CAD & CG, Zhejiang University

[5]New Media Technology & Art, Harbin Institute of Technology

Index Terms: [Human-centered computing]: visual analytics, geographic visualization, information visualization; [Information Systems]: spatial-temporal systems.

1 Introduction

The 2014 VAST Grand Challenge required us to find victims, suspects, and criminal motivations based on three separate datasets. We developed three VA tools (AnnotatedTimeTree, Dodeca-Ring Map and SMART) to facilitate the understanding of heterogeneous multivariate datasets. These tools were integrated to gain insights into the source data and find connections among complex information (Fig. 1). AnnotedTimeTree aims to identify the cause clues and timeline of kidnapping based on analysis of text and network data. Dodeca-Rings Map allows analysts to interact with geospatial, temporal, and card transaction data to find suspicious personal behaviors and social networks. SMART is a visual analytics tool that enables text stream analysis by dynamically visualizing microblog data on the map over time.

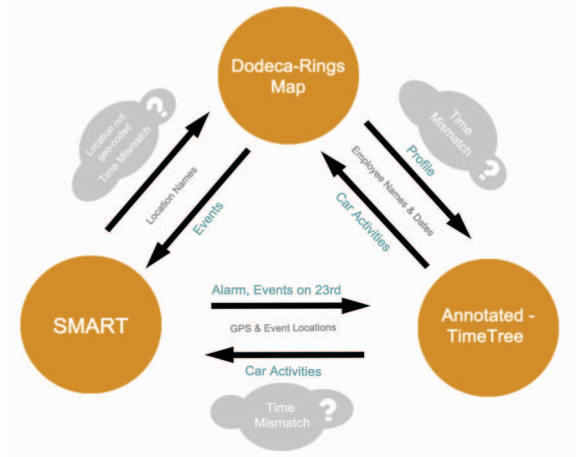


Figure 1: This Grand Challenges is solved by the collaboration of three visual analytics systems

To address this Grand Challenge, we collaborated widely with VA researchers from Purdue University, Harbin Institute of Technology, and Zhejiang University from China. The background of team members ranges from Computer Graphics Technology, Interaction Design, to Computer Engineering.

1.1 AnnotatedTimeTree – Mini Challenge 1

AnnotatedTimeTree is a time-critical document visualization tool designed for organizing large collections of reports and annotations. In order to capture the key events that happened in historical news and reports, we used a vertical tree-structure timeline to organize documents in reverse chronological order; the top ones are the most recent. The deep shade of green in each cell represents the numbers of documents. AnnotatedTimeTree also provides semantic interactions such as searching, highlighting, annotating, and expanding for maintaining reasoning process and reasoning results in the documents.

1.2 Dodeca-Rings Map – Mini Challenge 2

Dodeca-Rings Map is designed to analyze geo-temporal traffic problems. We used dodecagons that divide 24 hours into two-hour intervals to visualize one car's 24-hour activities. The color segments denote the stop periods in each location and color dots represent activities such as credit card transactions and twitter messages. The activity temporal chart opens all the Dodeca-Rings with the sample temporal length, and allows analysts to compare activities with different ordering choices. The social relationship matrix helps analysts to understand the general social connections.

1.3 SMART – Mini Challenge 3

SMART is a social media analytical tool to investigate temporal trend, geographical distribution, social networks, and semantic evolution. The top timeline presents statistical distributions of streaming data. The text clouds on the right side visualizes key words extracted from microblog data within filtered time ranges. On the background map of Abila City, blue dots show geo-tagged microblog data and the red dots represent call center data. The network view demonstrates the retweet or reply relationships. The analyst can use content lens to illustrate prominent keywords associated with selected microblog users.

2 Data Integration and Collaboration

Openness, connectivity, and flexibility are the key techniques we used for analyzing with the three tools. The original data sets of the three Mini Challenges have different formats. The Grand Challenge asked us to integrate network, text, temporal, geospatial, and stream data as well as explore the storyline of the disappearance incident. From the introduction, we can see that the whole story is a spatial-temporal event that involves strong underlying social networks. Thus we need to combine information gathered from all the three systems to make the final picture.

* email address: {guo171, yuj, jieqiongzhao, zhan1486, wang1925, qianz,victorchen@purdue.edu};xiajing@zjucadcg.cn;chwang@hit.edu.cn; ebertd@ecn.purdue.edu

IEEE Conference on Visual Analytics Science and Technology
2014 November 9 - 14, PARIS, FRANCE

IEEE Symposium on Visual Analytics Science and Technology 2014
November 9-14, Paris, France

978-1-4799-6227-3/14/$31.00 ©2014 IEEE

To identify the clues and investigate the events, we needed to allocate the information of key events' when (time), where (location), and who (the social network). In most cases the datasets only provided partial information of an event, for example, in MC3, we were give the location and time of standoff, but who stayed in the suspicious black van was not clear. Combining the partial information by matching the known pieces provided us with more information. Dodeca-Rings Map provided found that their houses received visits from security people in some very early mornings (Fig.2 A). Several security people routinely visited an executive's home at approximately 3:30am and then left at 8am. From the social relationship matrix, we also found that this group of security personnel has a close relationship, but are not friends with those executives. The suspicious early morning visits led us to form the hypothesis that these security personnel were involved in the kidnapping event.

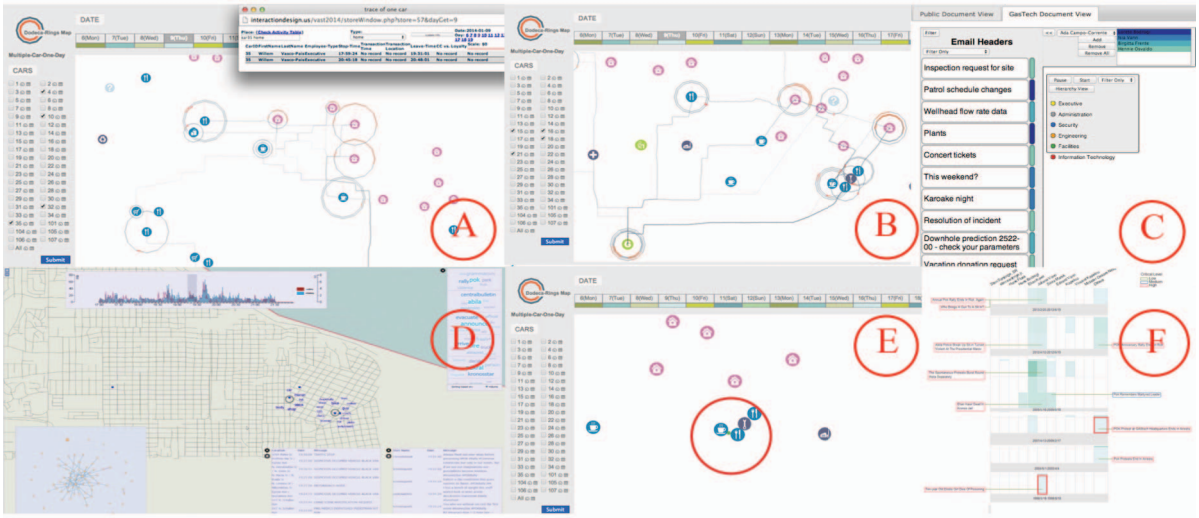


Figure 2: A: The executives' houses received visitations from security people in some very early mornings. B: Highlight several security people's names in Dodeca-Rings Map. C: Pass security people's names to AnnotatedTimeTree. D: Highlight the hit and run geolocation in SMART. E: Pass the suspicious location to Dodeca-Rings Map. F: POK has been protesting GasTech and Kronos Government for a long time.

spatial-temporal analysis and hints of several social networks within GAStech. Data from SMART and AnnotatedTimeTree contain rich spatial-temporal information about events. Linking up the events with networks is essential. AnnotatedTimeTree and Dodeca-Rings Map are connected by passing names between the two systems (Fig.2 A, B & C). Analysts can select employees' names in Dodeca-Rings Map and find relevant information in the document view of AnnotatedTimeTree, and vice versa. SMART and Dodeca-Rings Map are connected by passing geolocations (Fig.2 D & E). By clicking the suspicious geolocation in SMART, the same location will be marked on the Dodeca-Rings Map.

Dodeca-Rings Map has complete spatial-temporal information of the given two-week period. It can accept partial event information and provide a more complete picture of what happened before the disappearance incident. For example, given the person (car) and time of an event, the location of the event can be identified using Dodeca-Ring Map if the time is within the 2-week range. Similarly, given the person and location, the time of the event can be estimated at the period of the stay in the location.

3 Analysis Process

To investigate the events of January 20 related to the disappearance of the GAStech employees, we used the three visual analytics tools collaboratively. Analysts used AnnotedTimeTree and Dodeca-Rings Map to develop the initial hypotheses. SMART was able to provide evidence and support for current investigations and prove the various hypotheses.

At first, in the Dodeca-Rings Map system, we examined the executives' car activities in the multiple-car-one-day mode, and

By passing all the security people's names to the AnnotatedTimeTree's GAStech view, we found that two emails are significant. One was related to the protest of POK, and the other one attempted to obscure the motivation of the first email as a virus. These emails mentioned a person named Edward Vann, who shared the common family name with the key persons in POK. We also found that POK had been protesting GasTech and Kronos Government for a long time (Fig.2 F). Hence we assumed that the security department's employees, who were indeed POK members, kidnapped the executives.

Searching "kidnap" from SMART, we found that on January 23rd, a van hit a car and a bicyclist and then proceeded to head West on Egeou St. at approximately 7:30pm. By passing the hit and run geolocation to the Dedoca-Rings Map, we found that the place is near a car repair store called Frydos Autosupply n' More. Five security people frequently visited this store after work. The unusual activity further supported our assumption that several security people kidnapped four executives in GAStech on 20th.

4 Conclusion

The three visual analytics tools construct a powerful analysis system that helps analysts discover suspicious spatial-temporal patterns and behaviors in the data sets. Each system feeds information to others and helps to explore the unknowns. By passing different types of data among systems, we are able to retrieve and even create the missing temporal or geospatial information. For the future developments, we will build more direct linkages between the three systems and allow them to pass keywords, networks, archive documents, and filter information.

Chapter 28
Cross-Scale, Multi-Scale, and Multi-Source Data Visualization and Analysis Issues and Opportunities

David Ebert, Kelly Gaither, Yun Jang and Sonia Lasher-Trapp

Abstract As computational and experimental science have evolved, a new *dimension* of challenges for visualization and analysis has emerged: enabling research, understanding, discovery at multiple problem scales and the interaction of the scales, and abstractions of phenomena. Visualization and analysis tools are needed to enable interacting and reasoning at multiple simultaneous scales of representations of data, systems, and processes. Moreover, visualization is crucial to help scientists and engineers understand the critical processes at the scale boundaries through the use of external visual cognitive artifacts to enable more natural reasoning across these boundaries.

28.1 The Challenge of Multi-Scale Interactions

"Multi-Scale Interactions" has been used to characterize and emphasize that significant breakthroughs need to occur in a variety of fields by understanding both how the larger scales fuel the smaller scales, and how smaller scales feed back into larger scales. One fundamental example of this is turbulence. Turbulence is a major unsolved problem for fluid flow and is applicable to weather, medicine, engineering, and climate change. Biology gives us another example where scientists are working to understand structure and function from the cellular level up to the level of organs, then to functional subsystems within the body. Another example occurs in parameterization in numerical modeling: how one represents processes occurring at the

D. Ebert (✉) , Y. Jang · S. Lasher-Trapp
Purdue University, West Lafayette, USA
e-mail: ebertd@purdue.edu

S. Lasher-Trapp
e-mail: slasher@purdue.edu

K. Gaither
Texas Advanced Computing Center, Austin, USA
e-mail: kelly@tacc.utexas.edu

Y. Jang
Sejong University, Seoul, South Korea
e-mail: jjangyn@gmail.com

C.D. Hansen et al. (eds.), *Scientific Visualization*, Mathematics and Visualization,
DOI 10.1007/978-1-4471-6497-5_28

subgrid scale, and improving their parameterizations. If parameterizations are good, then the necessity to keep resolving smaller and smaller scales in one's numerical model disappears.

Another class of problems is filling data holes and gaps. How can new tools help us explore these holes and gaps between different kinds of observational data collected at different scales, or between the hierarchies of numerical models that we use to solve subsets of the problem that need to be merged to understand the relative contribution or importance of the solutions to the subsets?

28.1.1 Systems of Systems

The challenges of multi-scale interactions in complex systems and processes has led to new areas of research and refocusing of disciplines in engineering, as illustrated by the evolution of industrial engineering to industrial and systems engineering, and the development of the subarea of systems of systems research. Therefore, in order for these multi-scale interactions to be investigated and explored, tools are needed which scale to handle *systems of systems* [2]. These problems are common in science and engineering, and may require analysis and combination of data across scales. For example, macrobiology analysis may require understanding the interactions of data simultaneously at the genome, protein, cell, organ, human, country, and ecosystem levels. Cancer care treatment requires understanding and integrating data from the biomarker level (e.g., integrating metabolics, lipidomics, genomics, and proteomics data already at multiple scales) and cancer processes at the organ level, environmental exposure, and socioeconomic factors that affect the success and completion of treatment regimens.

Weather and the environment provide further examples, such as clouds and precipitation. Clouds and precipitation affect our daily lives, personal safety, commercial decisions, and our future sustainability on Earth. Clouds and precipitation are important at all regional scales: local, state, national, and global. Clouds influence the daily maximum and minimum temperatures over our homes and they modulate the global temperature by affecting the amount of incoming solar radiation and outgoing long wave radiation. Precipitation is likewise important at all scales. It directly affects our quality of life: our food supply, drinking water supply, air purity, modes of transportation, and many other human needs across the earth. As the inhabitants of earth become increasingly concerned about global warming and climate change on global and regional levels, it is necessary to understand the roles of clouds and precipitation in the Earth's System in order to predict the future state of our planet. However, fundamental questions remain concerning cloud motions and evolution, cloud longevity, and precipitation formation, and these gaps in our knowledge hamper our efforts to understand and predict weather and climate. Understanding and predicting clouds and precipitation are very difficult tasks which require the measurement and modeling of properties on a wide variety of scales (microscale, cloud scale, storm scale, mesoscale, synoptic scale, global scale as shown in Fig. 28.1), fusion of computational model data and measured data, and the simultaneous fusion of hundreds of

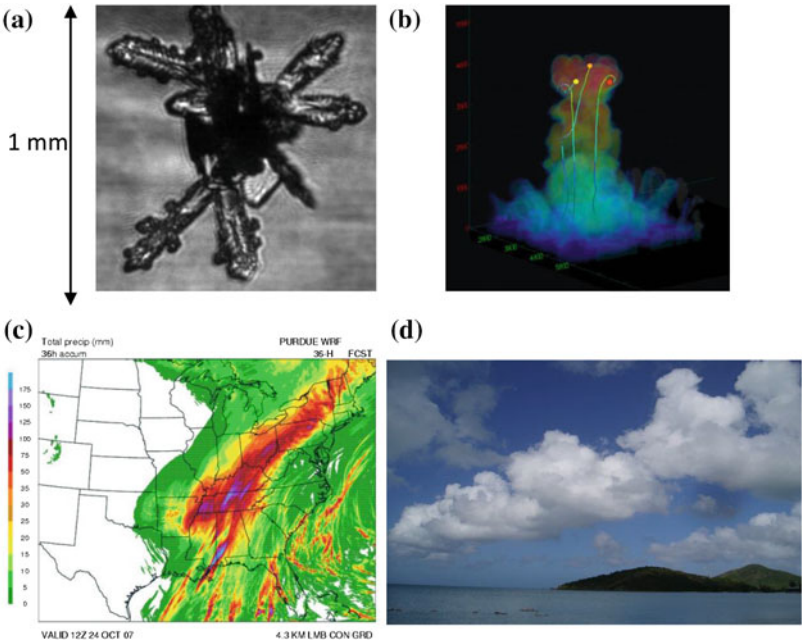


Fig. 28.1 Multi-scale examples, **a** microscale, **b** cloud scale, **c** storm scale, and **d** global scale. © IEEE reprinted, with permission, from IEEE transactions on visualization and computer graphics 12(5)

scalar and vector fields that vary over time. Current tools for atmospheric visualization and analysis are incapable of the following crucial functions:

- Integrating these various data sources and providing effective interfaces for fusion, analysis, experimentation, exploration, hypothesis testing, and discovery.
- Communicating the complex three-dimensional, time-varying information necessary to accurately predict atmospheric events, conditions, and consequences (e.g., aircraft icing) and extend the understanding of atmospheric phenomena.
- Integration of visual representations into the scientific analysis and discovery process.

28.1.2 Transformational Cross-Scale Science

Facilitating breakthrough scientific discoveries will require the development of transformational science that produces revolutionary new tools for mastering the multi- and cross-scale challenges of our world. Science discovery possibilities are presented here as a broad spectrum of disciplines, including the following.[1]

[1] This is summarized from the NSF Science and Engineering Community Workshop report by Ebert D., Gaither K., and Gilpin C.

- Computational Fluid Dynamics—Understanding computational fluid dynamics will allow us to understand and design a broad spectrum of applications: design more aerodynamic (and hence fuel-efficient) cars and planes, design better artificial hearts, perform better cardiovascular surgery, design better air conditioners, fans, and heat exchangers, better short-term weather prediction (including accurate precipitation forecasts and models), and understand the dynamics of global warming in the long term, improve our understanding of the dynamics of the oceans, and allow us to predict solar storms that affect radio communication, design better hydroelectric generators, design more efficient HVAC systems in buildings, design more efficient wind farms for the generation of electricity, improve the design of ship hulls, help us understand the mechanism of flight in birds and insects, and help us understand the locomotion of aquatic animals, including microorganisms.
- Preserving Coastal Margins—We have to preserve coastal margins so that our great-grand children will have access to a functioning environment that supports economic development and quality of life. By understanding cause-effect relationships between climate, human activity and coastal margins well enough to predict and communicate ecosystem evolution, we can effectively influence society's choice on affecting ecosystems health and sustainability.
- Virtual Paleoworld—We can reconstruct climate in the broadest sense for any time in the Earth's past and see what it looks like globally. This will allow us to see Tectonic plates in their proper shapes and positions and all data sites marked. We can see topography, heat flow, atmospheric composition, wind belts, biomes on land, and ocean currents as they existed then and with explicit depiction of differences from today and from individual data sites with accompanying uncertainties (e.g., model-data comparison).
- Understanding the Origin of Our Universe—Gravitational waves can probe to 10^{-43} seconds after the big bang. They carry key information of what happened just after it all began. Deciphering their content and comparing them with cosmological models will enable us to hone in on the model that captures reality. This will additionally have implications for a unified theory of physics, understanding dark matter and energy and why regular matter (like the one humans and stars are made of) only comprises about 3 % of the total.
- Understanding the 'Cradle of Life'—Supernovae are responsible for producing the needed energy that turns heavier elements (e.g., iron) of a soon to explode star into lighter elements that are the building blocks for life (hydrogen, oxygen, etc). Understanding such systems require complex simulations at the peta-(and beyond) scales and their solution have commonalities with other spectacular phenomena like gamma ray bursts. Obtaining the correct model to explain supernovae events will go a long way towards understanding what is required for the basic building blocks of life to be produced and explain the most spectacular astrophysical phenomena.
- Origin and Evolution of Languages—The origin and evolution of languages is a result of interactions with culture. We need to better understand how languages reflect the history of cultures, and how genetics/genomics data sets can be used to study unrecorded historical data concerning the migration of humans.

- The Cell at Subnanometer Resolution—The cell is the most basic building block of life. Cell function is a direct reflection of cell structure, and we understand the function of the cell only to the degree that we understand its molecular architecture. The molecular defects that underlie human disease are rooted in changes in the molecular components in the cell that alter cell structure and function. Molecular medicine would be greatly advanced by a molecule level view of the cell since it would guide the engineering of molecular therapies to fix what is broken in cells in the disease state.

28.1.3 Temporal Scalability

Sensemaking often involves temporal reasoning, and may require handling data at different time scales. For example, it may be necessary to understand long-term patterns by looking at data over a period of years or even decades, and simultaneously understand near-term effects by looking at data over a period of hours or less. Moreover, it may be necessary to integrate and perform correlative analysis on data collected at different temporal scales based on acquisition technology. For instance, in understanding fundamental principals of rain formation in clouds, it can be necessary to integrate data collected 1000 times per second with data collected every several minutes (radar data) and this information may then be fed into climate models that work on the scale of years and decades.

28.2 Variety of Data

Our ability to collect data is increasing at a faster rate than our ability to analyze it [3]. Scientists, engineers, and analysts are often overwhelmed with massive amounts of data from multiple sources and where the important information content exists in a few pieces. Therefore, we need to create new methods to allow them to visually examine this massive, multi-dimensional, multi-source, time varying information stream to make decisions more efficiently. The various data types include the following:

- Textual data—Massive textual data from documents, speeches, e-mails, or web pages now influence the problem domain. This data can be truly massive, contain billions of items per day, and much of it must be analyzed in a time-critical matter.
- Databases—Many corporate and government entities have constructed huge databases containing a wealth of information. We require new algorithms for the efficient discovery of previously unknown patterns in these large databases.
- Geospatial data—Consider the data collected by satellites that image the earth. We now have satellites that can create images at less than 1 m resolution and that can collectively image the land surface of the planet in a very short time. These images must be examined in a time-critical matter.

- Sensor data—The revolution in miniaturization for computer systems has allowed us to produce a myriad of sensors. The sensors can collect data about their environment (location, proximity, temperature, light, radiation, etc.), can analyze this data, and can communicate between themselves. Collections of sensors can produce very large streaming sets of data.
- Video data—Video analytics are being used more and more to enhance the effectiveness of the security in high-risk security operations. Content analysis, combined with massive recording capabilities, is also being used as a powerful tool for improving business processes and customer service. New techniques must be developed to integrate this streaming data paradigm into the analyst's toolbox.

Whereas each of these categories can produce massive data streams containing information that is applicable to a given problem domain, the grand challenge problem in the area of scalability is to use analytics to distill the relevant pieces of information from these widely disparate information streams, and create an information space containing relevant information that can be examined by analytical or visual means to influence the exploration, hypothesis testing, discovery, and decision making of the user. These systems need to provide mechanisms that can visualize the connections between the relevant information in the information streams, and allow the user to relate concepts, theories, and hypotheses to the data.

Several research directions present themselves as candidates to address these scalability problems, classified as visual scalability, information scalability, software scalability and information fusion.

28.2.1 Visual Scalability

Visual scalability [1] is the capability of visualization tools to effectively display massive data sets, in terms of either the number or the dimension of individual data elements. Factors affecting visual scalability include the quality of visual displays, the visual metaphors used in the display of information, the techniques used to interact with the visual representations, and the perception capabilities of the human cognitive system. A critical area of research in visual scalability is in methods that allow the user to change the visual representation of data.

28.2.2 Information Scalability

Information scalability implies the capability to extract relevant information from massive data streams. Methods of data scalability include methods to filter and reduce the amount of data, techniques to represent the data in a multiresolution manner, methods to abstract the data sets.

28.2.3 Software Scalability

A commonly held best practice in active data visualization begins with a visualization that summarizes a large data set followed by a subsetting to examine its detail. This practice requires active data visualization software that can execute visual queries and scale to data sets of varying sizes. Software scalability includes the generation of new algorithms that scale to the ever-increasing information sets that we generate today. We wish to avoid the hidden costs that arise when we build and maintain monolithic, non-interacting, non-scalable software models.

28.2.4 Information Fusion

Information fusion includes the capability to fuse the relevant information from divergent multi-source multi-dimensional time-varying information streams. This is the grand challenge problem in visualizations. Researchers must not just produce new visual representations and data representations for specific data types or information streams, but we must develop methods that fuse the relevant information into a single information space and develop new visual metaphors that allow the analyst to *look inside* this complex, multi-dimensional, time-varying space.

We must also develop techniques to *measure* scalability so new tools can be analyzed for their applicability in this domain. We must establish metrics that allow us to evaluate both visual metaphors and data representations as they apply to scalable algorithms. The best measurement will not only evaluate the representations according to scale, but also to the number of insights, actions, or value achieved for the analyst.

28.2.5 Technology Needs

What is needed in the visualization research agenda is to extend the state-of-the-art visual and data representations to be able to explore the heterogeneous multi-source multi-dimensional time-varying information streams. We must develop new visual methods to explore massive data in a time critical matter. We must develop new techniques for information fusion that can integrate the relevant pieces of information from multi-source multi-dimensional information. We must develop new methods to address the complexity of information, and create a seamless integration of computational and visual techniques to create a proper environment for analysis. We must augment our methods to consider visual limits, human perception limits, and information content limits. Therefore, the following challenges can have significant impact on science, engineering, discovery, and society:

- Develop quantifiable scalable visual representations, data representations, and software tools for various domains.
- Develop new methods for abstraction of massive streaming data from textual sources, satellite data, Sensor data, video data, and other information streams.
- Develop new research capabilities for information fusion. These methods should utilize visual analytic techniques to extract the relevant nuggets of information from heterogeneous multi-source multi-dimensional time-varying information streams, fusing these pieces into explorable information space.

References

1. Eick, S.G., Karr, A.F.: Visual scalability. J. Comput. Graph. Stat. **11**(1), 22–43 (2002)
2. Robertson, G.G., Ebert, D.S., Eick, S.G., Keim, D.A., Joy, K.: Scale and complexity in visual analytics. Inf. Vis. **8**(4), 247–253 (2009)
3. Thomas, J.J., Cook, K.A. (eds.): Illuminating the path: the research and development agenda for visual analytics. IEEE CS Press, NJ (2005)

Dodeca-Rings Map: Interactively Finding Patterns and Events in Large Geo-temporal Data

Chen Guo[1], Shang Xu[2], Jun Yu[1], Hanxing Zhang[4], Qian Wang[4], Jing Xia[5], Jiawei Zhang[3], Yingjie Victor Chen[1], Zhenyu Cheryl Qian[2], Chen Wang[4] and David Ebert[3],

[1]Computer Graphics Technology, [2]Interaction Design, [3]Electrical and Computer Engineering, Purdue University

[4]New Media Technology and Art, Harbin Institute of Technology, [5]State Key Lab of CAD & CG, Zhejiang University

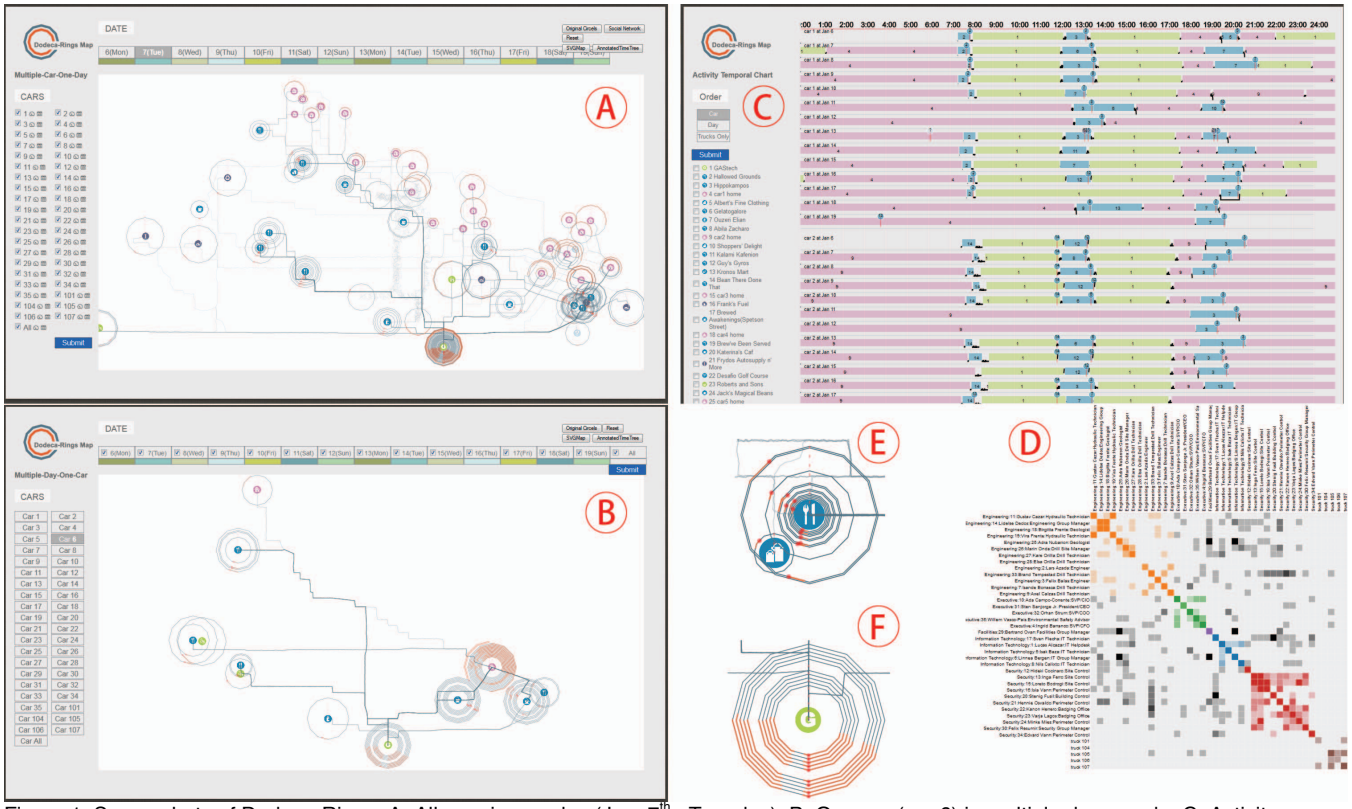


Figure 1: Screenshots of Dodeca-Rings. A: All cars in one day (Jan. 7th , Tuesday). B: One car (car 6) in multiple days mode. C: Activity temporal chart. D: Social relationship matrix. E: Dodeca-rings in Multiple-Car-One-Day mode. Red dots on the rings show transactions. Some cars love to come to this restaurant for lunch (around 12pm to 1pm) or dinner (around 8pm). F: Dodeca-rings in Multiple-Day-One-Car mode.

Index Terms: [Human-centered computing]: visual analytics, geographic visualization, information visualization; [Information Systems]: spatial-temporal systems.

1 Introduction

Dodeca-Rings Map is the visual analytics system we designed to analyze geo-temporal traffic problems. The system is organized by three kinds of visualizations: dodecagons that show events on the map (Fig.1 A & B), activity temporal charts (Fig.1 C), and a social relationship matrix (Fig.1 D). We used it to solve the VAST 2014 Mini-Challenge 2 and then the Grand Challenge. The given data sets include two weeks of vehicle GPS tracking data, credit and loyalty card transaction data, as well as vehicle assignments data. The challenge requires us to describe common daily routines for the car drivers, identify unusual events, and address the uncertainties and conflicts inherent in this data.

Dodeca-Rings map enables analysts to identify geo-temporal relationships quickly among the events on the map and provides rich interactive links to facilitate massive data analysis. According to Peuquet [1], three key components in spatial-temporal data are space (where), time(when), and objects (what). Dodeca-Rings map uses dodecagons as an innovative approach to integrate and visualize these three components in an elegant way. Each dodecagon visualizes the parking locations and time periods of stay for one car within one day. In the map, several dodecagons nested together to make concentric rings, which helps the analyst to see the patterns and events at first glance (Fig.1 E & F). The analyst can interact to zoom in and investigate the data. As the supplements in the system, activity temporal chart allows the analyst to read and compare events of all cars in all the days and the social relationship matrix provides a concluded social network overview. To support the visualization, we used different data

* email address: {guo171, xu537, yuj, zhan1486, victorchen, qianz}@purdue.edu; zhanghanxing1991@hotmail.com; wangqianmeiti@163.com; xiajing@zjucadcg.cn, chwang@hit.edu.cn

IEEE Conference on Visual Analytics Science and Technology 2014 November 9 - 14, PARIS, FRANCE

IEEE Symposium on Visual Analytics Science and Technology 2014 November 9 - 14, Paris, France

978-1-4799-6227-3/14/$31.00 ©2014 IEEE

mining techniques such as association, anomaly detection and clustering to extract information from the source data.

2 THE SYSTEM

Dodeca-Rings Map is a web application built on a standard LAMP (Linux, Apache, MySQL, and PHP) environment. It is composed of three key components: data storage layer, data process layer and data visualization layer. The provided datasets are stored in a MySQL database. PHP is used to access and manipulate data. HTML5, CSS, SVG, JavaScript, jQuery and D3.js are used to visualize data and provide interactivity. The data visualization layer is presented in modern web browsers. The integrated visualizations allow analysts to comprehend the dataset quickly and grasp large-scale patterns with minimal cognitive load.

2.1 Maps of Dodeca-Rings

The default view of our system is a geo-spatial visualization with maps of dodecagons that visualize cars' daily activities on the map (Fig.1 A, B). One dodecagon on the map visualizes one car's stays and activities in different locations in one day. The center of the dodecagon is clearly marked as the parking location. We differentiate these locations with numeric IDs and use different color logos to show their types (e.g. a house, an office, a restaurant, or a factory). The orange thick segments on the dodecagon indicate the periods of stay and color dots mark activities such as credit card transactions. The 12-sided polygon divides 24 hours into two-hour intervals. Thus the analyst can easily estimate when the events happened and the time length of stay period. Polylines connecting centers of dodecagons show the cars' traces from one place to another. We can turn on or off the background map to display roads. According to the original GPS data, we observed that cars were parked closely in clusters since they should be normally parked in parking lots. We grouped these locations into common centers. Two locations will be merged as one if their distance is less than 50 meters. As a result, on one dodecagon, it is possible to have multiple segments and activity points to show that the car has stayed in this location for multiple time periods. For some missing information, this system also allows the analyst to detect and assign different names and types to different locations during the analysis process. We also use the radius of the dodecagon to encode the car ID or the date. Different radiuses show different cars or dates, so several dodecagons can be nested together to become co-centric rings. Depending on the encoded data, the dodeca-rings view has two modes: one shows multiple cars' activities in one day (Fig.1 E) and the other shows one car's activities in multiple days (Fig.1 F).

2.1.1 Multiple-Car-One-Day Mode

In this mode, from inside to outside, different radiuses of the rings represent vehicle IDs from car 1 to car 35 and 5 trucks. With this visualization, the common patterns of cars in one day are obvious: leaving their homes, having breakfast, going to office, leaving for lunch, returning to office, and dining out for dinner. In this view, the analyst can highlight, select and compare multiple cars' activities through mouse over and clicking. Mouse over on one dodecagon will cause the system to highlight all other dodecagons of the same car. Clicking on the location icon shows the detailed information of people and their activities of that day in a summarized table. We can compare credit card charges and loyalty card records simply with colored lines. In this mode, we can easily see events that involve in many cars to go to the same place at the same time. Also we found that some car drivers have very close relationship since they tend to go to the same restaurant at the same time, or frequently participate in the same event.

2.1.2 One-Car-Multiple-Day Mode

In this mode, from inside to outside, the rings indicate days from January 6th to 19th (Fig.1 B). With this visualization, one car's weekly patterns are clear. For example, car 1 came to office at 8 am and left at 5:30 pm during most of the weekdays. But on 6th and 9th, he came to the office at midnight. During the weekend, the car did not come to the office at all. Interaction in this mode is similar to the previous mode. Mouse over one dodecagon will high light other dodecagons of the same day. Thus the analyst can select and compare individual days. Clicking on the location icons will show all the detailed activities of the car in a summarized table.

2.2 Activity Temporal Chart

The activity temporal chart (Fig. 1 C) is an opened-up variation of the dodeca-rings. It compares cars' activities through rows of colored bars. Each row represents details of one car in one day. The analyst can read the car speed, card transactions, and stopped locations. Car stops are marked in bars with colors that are consistent with dodeca-rings. Activities such as credit card transactions are marked as color drops. The analyst can easily see problematic activities that do not match the car locations. The speeds of cars are visualized as black curves connecting the car stays. It is very obvious that there is no speeding record for all the cars in the two weeks. The analyst can choose different ordering sequences to compare activities. When ordering by cars, analysts can see the car's daily patterns for the two weeks and compare different days of the car. When ordering by day, analysts can see all cars activities in that day and compare different cars. When ordering by truck IDs, analysts can identify which driver drove which truck through matching the card holder's credit card transaction with the truck stop locations.

2.3 Social Relationship Matrix

The social relationship matrix (Fig. 1 D) is a summary of general social connections of all the GAStech vehicle drivers. Different types of occupations are coded in different colors. If two drivers stayed at the same location during the same period of time, we consider the two cars were meeting. Maybe one or two meetings are coincidence, but frequent meetings can definitely imply interaction. The darkness of cell is defined by the number of meetings. For example, it is obvious that several security employees and one engineer are engaging in multiple meetings.

3 INTERACTION

The two modes of dodeca-ring map, activity temporal chart and the social relationship matrix are closely linked together in the system. The analyst can switch across two modes, pan and zoom into regions of interest to see details of suspicious patterns, highlight dodecagons of selected cars or days, select rings to compare, and open up rings to charts for further investigation. Although there are errors and missing parts in the datasets, the system provides functions for the analyst to manipulate the data, for example, enter and modify location names and types based on the credit card transaction data. Combined with various visualization and interaction methods, the dodeca-rings gives us sufficient support to analyze the VAST 2014 challenge.

REFERENCES

[1] D. J. Peuquet, "It's About Time: A Conceptual Framework for the Representation of Temporal Dynamics in Geographic Information Systems," *Ann. Assoc. Am. Geogr.*, vol. 84, no. 3, pp. 441–461, 1994.

Learning and Law Enforcement: How Community-Based Teaching Facilitates Improved Information Systems

Kaethe Beck
Purdue University
kaethe@purdue.edu

Scott Beamon
Indiana State Police
sbeamon@iifc.in.gov

Edward Delp
Purdue University
ace@purdue.edu

David Ebert
Purdue University
ebertd@ecn.purdue.edu

Abstract

To keep pace with the dynamic environment of information systems, it's necessary to prepare the next generation of the workforce for entry into this atmosphere. Department of Homeland Security Center of Excellence: VACCINE has partnered with the INGang Network, a component of the Indiana Intelligence Fusion Center to facilitate the best preparation for students and assist with some information system issues in the law enforcement industry. Exposing students to real-world applications not only facilitates the problem-solving process with respect to information systems, it also makes the student aware of the prevalent system issues. By participating in community-based teaching with law enforcement officers, the students gain a better understanding of the end user which allows them to design better information systems. There is an initial learning curve associated with integrating in any community, having a close working relationship with INGang has lowered the barrier to entry, ultimately allowing for a better information system to be created.

1. Introduction

Established in July of 2009, the Visual Analytics for Command, Control, and Interoperability Environments Center (VACCINE), along with its co-lead, Rutgers University, has served as the Department of Homeland Security's (DHS) Center of Excellence in Command, Control and Interoperability. VACCINE's mission focuses on creating methods and tools to analyze and manage vast amounts of information for all mission areas of homeland security, including our first responders and law enforcement officials. VACCINE accomplishes its mission through an integrated program of research, education and outreach, spanning the disciplines of visualization and computer graphics, engineering, computer science, geographic information systems, cognitive psychology, information technology, and emergency management and public safety. In pursuit of the center mission, VACCINE has partnered with over 40 first responder and law enforcement entities in order to ensure that our software tools enhance their ability to obtain, process, and gain insight from information. This problem driven research approach not only ensures real-world solutions and enhances transition to practices, but also provides greater research challenges. Our approach also greatly enhances the educational experience of our students; in initiating these partnerships, the graduate students of VACCINE are provided the opportunity to witness first-hand, the struggles with information systems currently in place for first responders and law enforcement officers. While VACCINE has numerous partners engaged in this problem-driven research, the Indiana Fusion Center and Indiana Gang Intelligence Network (INGang Network) have worked closely with our students to explain the struggles associated with information sharing and solution development.

VACCINE focuses on the research, development, and deployment of interactive visual analytic environments for communicating and disseminating information and deriving insight from the massive data deluge. Visual analytics is defined as the science of analytical reasoning facilitated by interactive visual interfaces[1]. Part of the mission of the center is to help decision makers make sense of the sea of text, sensor, audio, and video data by developing powerful analytical tools and interactive visual decision making environments that enable quick, effective decisions as well as effective action and response based on available resources. VACCINE integrates data and analysis into interactive visual displays to enable users to make discoveries, decisions, and plan action; in the project with the INGang network, that interactive visualization and analysis tool took shape in the form of GARI.

2. Background

While community-based teaching is nothing new, the concept is being applied to novel fields and disciplines such as software engineering and design[2]. Historically, community-based teaching

978-1-4799-2504-9/14 $31.00 © 2014 IEEE
DOI 10.1109/HICSS.2014.609

has most often been seen in the medical field. In 1993, *Tomorrow's Doctors* recommended an increase in community based teaching for undergraduates entering the medical profession[3]. This endorsement in correlated to a change in the methodology of preparing students for an occupation in the medical field, and is now common practice.

In applying this same approach to software engineering and system design, similarly to medical students, the individuals learn how to cope with patients or in this case, end users. Students routinely report they enjoy the community based teaching and feel they are better prepared for the workforce. In providing the experience of an internship without the time constraints, students can continue progress towards their degree, while gaining the real-world experience so many corporations and entities value – and for good cause. The practical application the students experience during this community based teaching requires that they increase their communication, leadership, and interpersonal relationship skills. In the field of software engineering or computer science, this can mean the difference when competing for a sought after position. Many universities require a speech or communications course in the engineering and computer science fields in an attempt to secure this skill among graduates whose stereotype is reserved and more comfortable communicating through machines. There is virtue in the ability to explain technical jargon in a language that individuals with a non-technical background can understand. Being able to speak technically and explain difficult concepts on a general level is a coveted skill. In participating in community based learning with the INGang network, these students have built their skill set to be better communicators, leaders, and ultimately, better system designers.

VACCINE has developed a strong partnership with the Indiana State Police, the Indiana Fusion Center, and in particular with the INGang Network. The INGang Network, which is run through the state fusion center, was developed in order to have a cohesive network to share information regarding gangs and gang activity among law enforcement officers. The creation of INGang is a perfect example of the issues that law enforcement officers encounter with respect to information systems and why it is critical to train the future system designers well through community based instruction.

Knowledge and information transfer among law enforcement officers is not facilitated by the current databases and information systems available. As such, the Indiana Fusion Center and State Police created a network of individuals dedicated to sharing gang related information. The creation of INGang is for the sole propose of transferring and propagating information. This is a prevailing theme across law enforcement agencies. There is no standardized database or record management system; each department or entity has selected their own, some have even custom designed systems to meet their individual needs. VACCINE graduate students have the opportunity to work with the law enforcement officials to assist them with their information systems and in particular, extracting information from them in a useful, productive manner.

3. Related Work

The concept of community-based teaching is common in a number of disciplines such as education or medical professions – for good cause. It is a common belief that there is no better training than experience. It is, however, not unheard of in other science disciplines. Purdue University's College of Engineering has created a program called EPICS – Engineering Projects in Community Service. The idea of EPICS is to provide students an opportunity to see how their work can impact the community, and teach them something along the way[4]. The projects can be quite long term (up to 3.5 years in some cases), and are almost always multidisciplinary in nature. The program went national in 1997 in order to reach a wider range of students. In this pursuit, Butler University attempted to adapt the current structure of the program in 2001 to apply to Computer Science, specifically for the field of Software Engineering (a group of students who are likely to develop information systems in the future)[5]. Butler was in the process of creating a software engineering degree and they wanted the students to have the opportunity to act with real customers on a deeper level than an internship would provide. They also wanted this relationship to be a longer-term time commitment than internships traditionally last. In this manner, the students could experience what a career in software development would mean.

After completing the first year of this program at Butler, the students were surveyed as a method of determining the effectiveness of community-based teaching. As expected, they found the majority of the students valued the experience – 92% even found that the EPICS program had a substantial impact on their customer awareness[5]. This is significant in the field of information systems. In order to design a productive, successful information system, understanding the end user or customer base is a critical step. If students feel that community-based teaching makes them more aware of the end user, that is one step closer to developing a better information system.

The original EPICS program at Purdue University also surveys its students regularly. In 2012 a survey was distributed to every alumnus that had a registered address in the school's database. Of the 2500 or so solicitations to participate, 528 surveys were completed. The alumni who participated varied in their majors and number of semesters involved in the EPICS program. From the survey, more than 70% of the alumni felt that the EPICS program had "some, large, or very large" impact on that individuals performance as an employee once they entered their professional field. Additionally, 20% claimed that EPICS actually influenced their selection of a career and went on to explain how[4].

4. GARI

The INGang Network and VACCINE have entered a two year pilot program in order to collaboratively develop tools designed to facilitate the transfer of information among the INGang Network. Our initial tool in its testing phase is the Gang Graffiti Automatic Recognition tool, or GARI[6]. This application was designed to catalog and categorize gang graffiti images. Gangs and gang violence are a major issue across the country. With some 33,000 violent gangs encompassing 1.4 million members, the ability to convey information among law enforcement officials is a necessity[7]. Gang graffiti has unique symbols, colors, and methodology to indicate threats, meetings, or other messages. By creating a tool which can essentially be used as a repository for information to decode gang graffiti, VACCINE students and law enforcement officers have created a new information system to facilitate this transfer of knowledge. VACCINE students were integral to designing the system. In working closely with INGang, the students were able to experience for themselves the struggles and frustrations of the current information systems and better understand the issues facing end users.

After learning of the issues with sharing and categorizing information related to gang activity, the students could see that the current information system would not suffice to provide the appropriate information and, moreover, would not allow for that information to be easily shared among law enforcement officials. GARI allows for the capture of graffiti images on Android and iPhones (expected late summer 2013). The application then uploads the images to the GARI server that, if so requested, will run an algorithm to test the image against other known images in the system in order to find similar images. Each image can be tagged with various levels of information – the meaning of a symbol, the associated gang, officer comments or notes, etc. The tool will then allow any other member of INGang who has installed the software to view the images and annotation. There is a map tool for looking at the geographic distribution and placement of graffiti – the software will even use the GPS location of the phone to display the graffiti images within a selected radius.

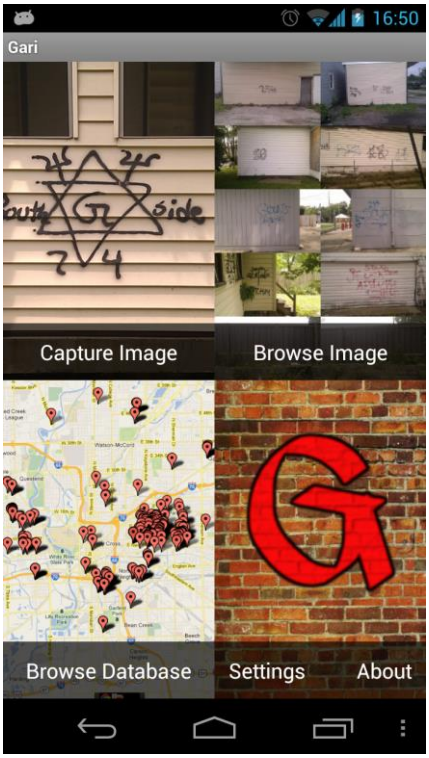


Figure 1: GARI app main screen

5. Student Experience

In order to look at the future of information systems, it's critical to look at the individuals who will be developing those systems. VACCINE students have a unique experience in their education as they are expected to produce a deliverable and deal with an end user/customer, in this case the INGang Network. These very same students will recall the frustration and issues they encountered and overcame when examining the different record management systems from county to county, or attempting to install a program designed to tunnel to a server on a secure network. These experiences have made them aware of the common and prevalent issues that propagate throughout information systems that are critical to safety.

6. Conclusion

Based on the experience in interacting with the INGang network in combination with research related

to community-based teaching, the approach appears to be an excellent method of preparing students for issues they will encounter and could avoid by appropriately designing information systems. As information systems can determine so many facets of productivity in the workforce, it behooves educators charged with preparing the next generation of professionals, to find the best method of educating and training students to develop and design these systems. The VACCINE partnership with the INGang Network arose from the need for information sharing outside of the existing systems, as they were not well designed. By allowing the students to facilitate a new system, the INGang network has also trained a better information system expert. This initial step is the start of the a Midwest partnership that will explore challenges at agencies of various sizes and structures in order to ensure the students are well-rounded in their exposure to the information systems available in the law enforcement arena while exposing them to any number of issues from a lack of network to severely limited bandwidth. With this increased exposure and practical experience, the students are well prepared for a globally competitive, highly distributed workforce.

7. Acknowledgements

This work is supported by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0003.

8. References

[1] J. J. Thomas, K. A. Cook, and National Visualization and Analytics Center, *Illuminating the path*. Los Alamitos, Calif.: IEEE Computer Society, 2005.

[2] J. A. Cantor, "Experiential Learning in Higher Education: Linking Classroom and Community. ASHE-ERIC Higher Education Report No. 7.," ERIC Clearinghouse on Higher Education, Graduate School of Education and Human Development, The George Washington University, One Dupont Circle, Suite 630, Washington, DC 10036-1183 ($18, plus $3.75 postage and handling)., ISBN-1-878380-71-0, 1995.

[3] K. Coleman and E. Murray, "Patients' views and feelings on the community-based teaching of undergraduate medical students: a qualitative study," *Fam. Pract.*, vol. 19, no. 2, pp. 183–188, Apr. 2002.

[4] J. L. Huff, W. C. Oakes, and C. B. Zoltowski, "Work in progress: Understanding professional competency formation in a service-learning context from an alumni perspective," in *2012 Frontiers in Education Conference Proceedings*, Los Alamitos, CA, USA, 2012, vol. 0, pp. 1–3.

[5] P. K. Linos, S. Herman, and J. Lally, "A service-learning program for computer science and software engineering," in *Proceedings of the 8th annual conference on Innovation and technology in computer science education*, New York, NY, USA, 2003, pp. 30–34.

[6] "The Graffiti Code Breaker | DiscoverMagazine.com," *Discover Magazine*. [Online]. Available: http://discovermagazine.com/2012/sep/25-the-graffiti-code-breaker#.Ub0qFvm1G6M. [Accessed: 16-Jun-2013].

[7] "2011 National Gang Threat Assessment," *FBI*. [Online]. Available: http://www.fbi.gov/stats-services/publications/2011-national-gang-threat-assessment/2011-national-gang-threat-assessment. [Accessed: 16-Jun-2013].

PLOS ONE

RESEARCH ARTICLE

Mass Media and the Contagion of Fear: The Case of Ebola in America

Sherry Towers[1]*, Shehzad Afzal[2], Gilbert Bernal[1], Nadya Bliss[1], Shala Brown[1], Baltazar Espinoza[1], Jasmine Jackson[1], Julia Judson-Garcia[1], Maryam Khan[1], Michael Lin[1], Robert Mamada[1], Victor M. Moreno[1], Fereshteh Nazari[1], Kamaldeen Okuneye[1], Mary L. Ross[1], Claudia Rodriguez[1], Jan Medlock[3], David Ebert[2], Carlos Castillo-Chavez[1]

1 Arizona State University, Tempe, AZ, U. S. A., **2** Purdue University, West Lafayette, IN, U. S. A., **3** Oregon State University, Corvallis, OR, U. S. A.

* smtowers@asu.edu

Citation: Towers S, Afzal S, Bernal G, Bliss N, Brown S, Espinoza B, et al. (2015) Mass Media and the Contagion of Fear: The Case of Ebola in America. PLoS ONE 10(6): e0129179. doi:10.1371/journal.pone.0129179

Academic Editor: Christos A. Ouzounis, Hellas, GREECE

Received: November 17, 2014

Accepted: May 5, 2015

Published: June 11, 2015

Data Availability Statement: All relevant data are within the paper and its Supporting Information files.

Funding: The parameter optimization portion of the analysis was made possible with education allocation #DMS140043 of super-computing resources from National Science Foundation Extreme Science and Engineering Discovery Environment (XSEDE) high-performance computing initiative (ST GB NB SB BE JJ JJG MK ML RM VMM FN KO MLR CR CCC). This research was also partially supported by the Western Alliance to Expand Student Opportunities (WAESO) Louis Stokes Alliance for Minority Participation (LSAMP) Bridge to the Doctorate (BD) National

Abstract

Background

In the weeks following the first imported case of Ebola in the U. S. on September 29, 2014, coverage of the very limited outbreak dominated the news media, in a manner quite disproportionate to the actual threat to national public health; by the end of October, 2014, there were only four laboratory confirmed cases of Ebola in the entire nation. Public interest in these events was high, as reflected in the millions of Ebola-related Internet searches and tweets performed in the month following the first confirmed case. Use of trending Internet searches and tweets has been proposed in the past for real-time prediction of outbreaks (a field referred to as "digital epidemiology"), but accounting for the biases of public panic has been problematic. In the case of the limited U. S. Ebola outbreak, we know that the Ebola-related searches and tweets originating the U. S. during the outbreak were due *only* to public interest or panic, providing an unprecedented means to determine how these dynamics affect such data, and how news media may be driving these trends.

Methodology

We examine daily Ebola-related Internet search and Twitter data in the U. S. during the six week period ending Oct 31, 2014. TV news coverage data were obtained from the daily number of Ebola-related news videos appearing on two major news networks. We fit the parameters of a mathematical contagion model to the data to determine if the news coverage was a significant factor in the temporal patterns in Ebola-related Internet and Twitter data.

Conclusions

We find significant evidence of contagion, with each Ebola-related news video inspiring tens of thousands of Ebola-related tweets and Internet searches. Between 65% to 76% of the variance in all samples is described by the news media contagion model.

Science Foundation (NSF) "Multidisciplinary STEM Solutions LSAMP Bridge to the Doctorate" grant #HRD-1401190 (GB SB JJ JJG MLR CR), and the Offices of the President and Provost of Arizona State University. This work was funded in part by the U.S. Department of Homeland Security (DHS) VACCINE Center award #2009-ST-061-CI0001-06 (SA DE), and was also made possible by grant #1R01GM100471-01 from the National Institute of General Medical Sciences (NIGMS) at the National Institutes of Health (CCC). The contents of this manuscript are solely the responsibility of the authors and do not necessarily represent the official views of the funding agencies. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

Introduction

Use of Google search trend data to predict disease outbreaks was first proposed in 2006 [1], and subsequently developed further in the years prior to the 2009 A/H1N1 outbreak [2, 3]. This resulted in the advent of the Google Flu Trends online application, which uses a combination of flu-related Internet search activity to predict, in real-time, flu outbreaks in the U. S. and other countries worldwide (see http://www.google.org/flutrends/, accessed February 9, 2015).

Twitter is a social networking and micro-blogging service that enables its millions of users to send and read each other's tweets, consisting of short, 140-character messages. The service currently has over 280 million monthly active users, sending on average around 500 million tweets per day (see http://about.twitter.com/company, accessed February 9, 2015). Twitter was first proposed as a means to track disease in real-time in a 2011 publication that looked retrospectively at flu-related tweets during the 2009 A/H1N1 pandemic [4].

Since these initial analyses, the field of "digital epidemiology" has rapidly expanded [5–8]. For example, Twitter data has been used to examine cholera outbreaks [9], vaccination sentiments [10], and proposed for use for global infectious disease surveillance [11]. Google search data has been used for real-time forecasting of Dengue outbreaks [12, 13], Listeria outbreaks [14], the spread of tuberculosis [15], consumer behavior [16], and the economy [17].

However, there have been some failings of these methods, leading to some concern about their applicability in an outbreak of an emerging infectious disease. At the beginning of the 2009 A/H1N1 outbreak, Google Flu Trends was not accurate in predicting the progression of the pandemic, due to information seeking behavior that was likely induced by mass media coverage [18]. Indeed, Chew et al (2009) found that some peaks in flu-related Twitter activity during the 2009 A/H1N1 pandemic were correlated to the timing of major news stories about the pandemic [19]. Additionally, during the severe 2012–13 influenza season, Google Flu Trends did not accurately predict the course of the outbreak, seriously overestimating the disease burden in the population. It has been suggested that the problems may have been due to widespread media coverage, including the declaration of a public-health emergency by New York state [20]. It has also been noted that news media is effective in "pulsing" of public opinion during election times [21], as expressed on social media.

Despite the fact that evidence of the influence of news media on Twitter and Internet search patterns has been repeatedly observed, up until now it has been extraordinarily difficult to correct for this effect because the patterns are also intertwined with temporal patterns due to the other dynamics of interest, such as the actual spread of a disease within a population.

During the recent very limited U. S. Ebola outbreak, the popularity of Ebola-related Google searches in the U. S. rivaled that of flu-related searches during the 2009 A/H1N1 pandemic (see Fig 1), but we can be confident that none (or virtually none) of the Ebola-related U. S. Internet searches or tweets arose from actual victims of Ebola in the U. S. The situation thus provides an excellent means to determine how public interest, curiosity, or panic regarding a certain topic affects social media and Internet search dynamics, and allows us to examine the influence of news media on these trends. The results of this study will thus help to inform future digital epidemiological analyses of outbreak data, possibly allowing for the correction of the effects of benign interest, information-seeking behavior, or public hysteria.

In this analysis we employ a mathematical model of contagion to simulate the potential influence of Ebola-related news videos on peoples' tendency to perform Ebola-related Internet searches or tweets. The advantage of such a model over the use of a simple statistical regression model to explore these relationships is that a mathematical model can incorporate non-linear dynamics related to people becoming immune to news media induced interest in a topic (i.e. the model incorporates the effect of eventual boredom with a topic, no matter how many news

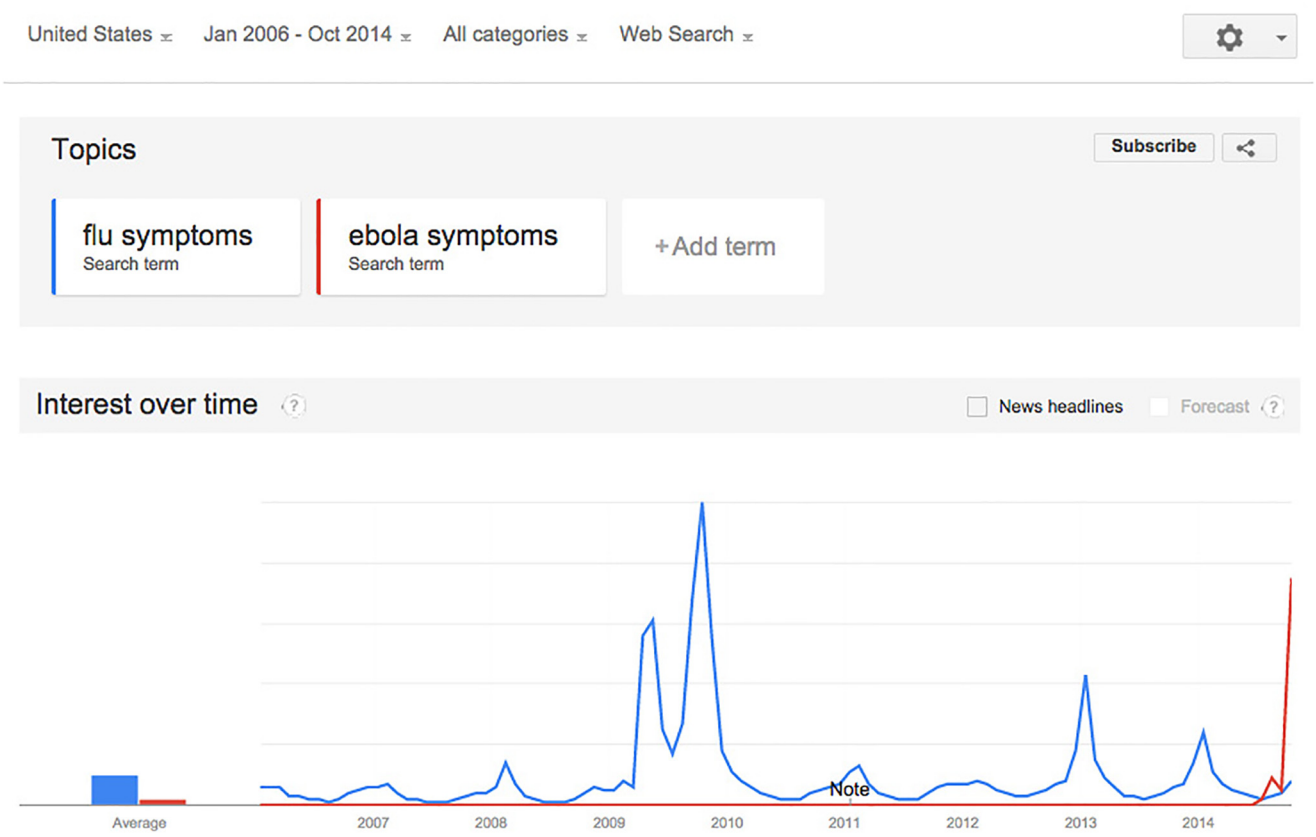


Fig 1. Comparison of the 2014 Ebola-related Google search trends to influenza-related search trends during the 2009 A/H1N1 pandemic. The relative interest in Ebola-related searches during the month of October 2014 rivaled the flu-related searches at the beginning of the A/H1N1 pandemic.

doi:10.1371/journal.pone.0129179.g001

videos are subsequently aired). For comparison purposes, however, we also present the results of statistical linear regression models examining these relationships.

As a cross-check to our analysis, we verify that public interest in Ebola (as reflected by Internet searches and tweets) does not appear to induce news media stories on the topic. We also show that peaks and valleys in the temporal trends in news media coverage tend to precede the same features in the Internet and Twitter data, not the other way around.

In the following sections we describe the sources of data used in this analysis, and describe the contagion model employed, followed by a presentation of results and discussion.

Materials and Methods

Data

Google search data. Daily trends in Ebola-related Google searches were obtained from Google Trends application program interface (API) (http://www.google.com/trends). The search terms examined were:

"Ebola"

"Ebola symptoms"

"Do I have Ebola"

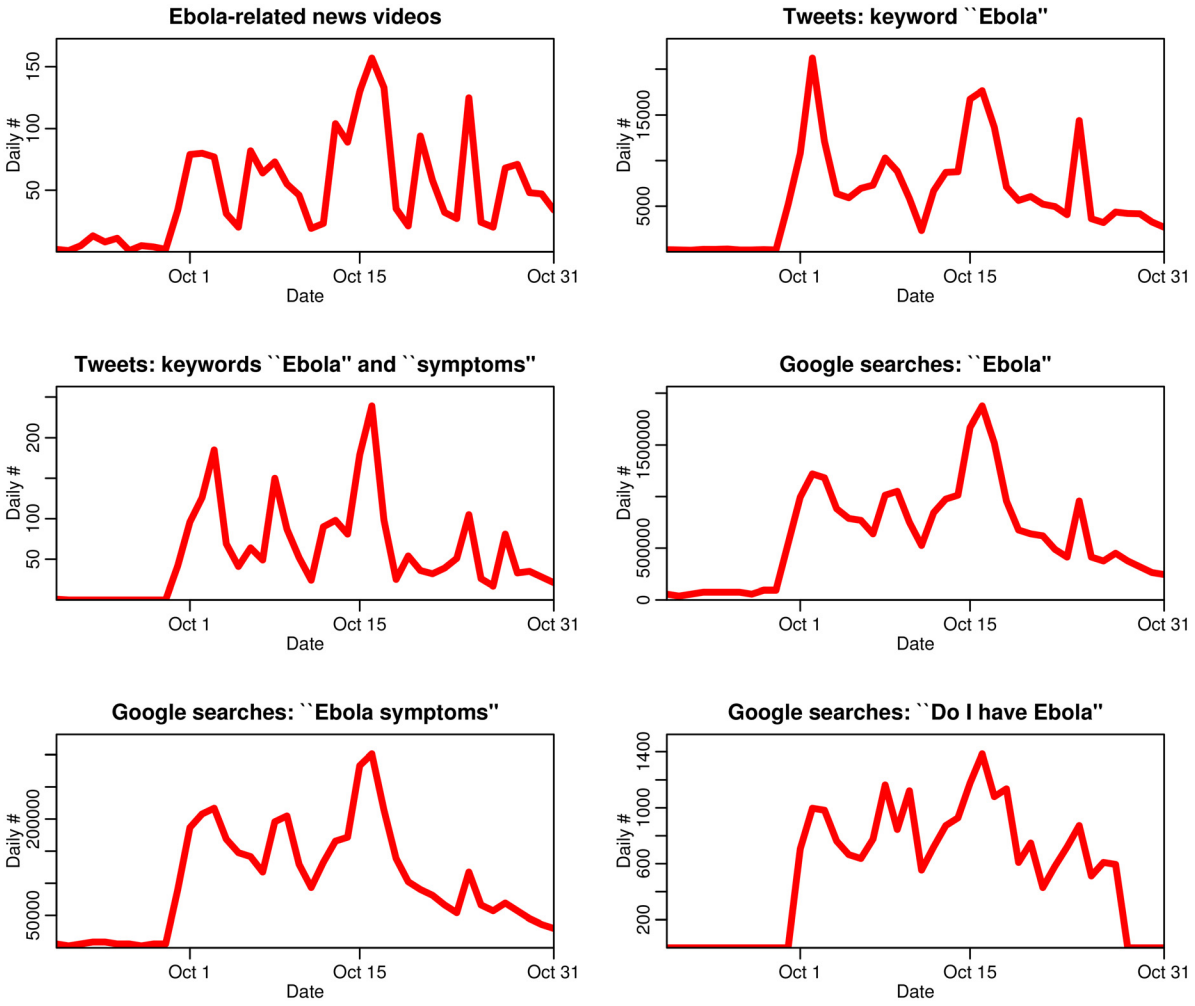


Fig 2. Time series of Ebola-related news media, Twitter, and Google search data used in this study. The samples consist of six weeks of data ending October 31st, 2014. The first case of Ebola confirmed in the U. S. occurred on September 29, 2014. The temporal trends in the data samples are highly inter-correlated, with a minimum of 70% correlation between samples.

doi:10.1371/journal.pone.0129179.g002

These search terms were chosen to reflect varying degrees of interest in the topic, from casual curiosity to possible panic regarding the potential of personal infection with Ebola. Only searches originating in the U. S. were considered for this analysis.

The data provided by Google Trends are normalized to the total search volume on Google, and thus the data are relative and not absolute. To estimate the total number of searches, we obtained the total number of searches per month originating in the U. S. for each search term from the Google Adwords subscription service, via the Keyword Planner tool, and scaled the Google Trend data accordingly.

The data are shown in Fig 2.

Twitter data. Daily Twitter data were obtained from a repository continuously collected by a server maintained by the Purdue center for Visual Analytics for Command, Control, and Interoperability Environments (VACCINE), which uses the Twitter streaming Application Programming Interface (API) to access up to 1% of the global stream of tweets.

From this stream, we selected tweets during the six week period ending October 31, 2014 containing the keyword “Ebola”, or the keywords “Ebola” and “symptoms”. Only English

language tweets geo-located in the United States were considered, and retweets were removed from the sample, resulting in a total of 260947 tweets.

Potential bias in the temporal trends due to tweets coming from news agencies, rather than regular (aka "grassroots") Twitter users needs to be minimized. We thus separated the tweets coming from obvious news agencies, public entities, or reporters; CNN, NBC, ABC, CBS, FOX, MSNBC, New York Times, Washington Post, Reuters, Bloomberg News, New Day, U.N. Spokesperson, and Geraldo Rivera. We also excluded tweets coming from users with usernames that included the words "TV", "media", "trend", "CP" (which often stands for "City Press"), and "report" (all selections were applied such that they were not sensitive to the case of the characters in the username). This yielded a sample of 10224 tweets.

As noted in Reference [22], the Twitter accounts of news agencies compared to the accounts of grassroots users tend to have a much smaller indegree to outdegree ratio (i.e. they usually follow far fewer accounts than they have followers). Reference [22] found that this ratio was on average 0.5 for news agencies (with median 0.001), and between 1.1 to 2.6 for non-media Twitter users (with median between 0.8 to 1.6). For our particular sample, we find this ratio is 0.34 for news agency users (with median 0.007), and 1.3 for the remaining users (with median 0.95). Thus, based on both the user name, and the properties of the user account, the remaining 250723 tweets are consistent with having not come from news agencies. These remaining data are shown in Fig 2.

News video data. According to a 2013 Gallup poll of over 2,000 people, 55% of Americans turn to television to obtain their news [23], with the next most popular medium being the Internet (22%). According to another study done in 2014 by the Pew Research center [24], CNN, MSNBC, and Fox News are the three primary cable news networks, with Fox News and MSNBC having an 80% share of that market.

While CNN makes available online a repository of its very recent videos, they are not reliably identifiable by date. However, both Fox News and MSNBC provide tools on their websites to obtain videos related to specific topics within specific date ranges. These videos are aired both on television, and also available for online viewing. We use these data as a proxy for the temporal trends in the total amount of news media coverage related to Ebola.

From these repositories, we downloaded the daily number of Ebola-related videos for the six week period ending October 31st, 2014. The data are shown in Fig 2.

Granger Causality

The Granger causality test is a hypothesis test for determining whether or not one time series is able to forecast another [25]. Granger causality is defined on the principle that a cause necessarily must happen prior to its effect, and that the cause contributes unique information regarding the future value of its effect.

A time series X is said to Granger-cause Y if it can be shown that X provides statistically significant information about the future values of Y. This is achieved through F-tests based on linear regression of Y on lagged values of X (including also lagged values of Y). Here, we examine the potential Granger causality of temporal patterns in the daily number of news videos on temporal patterns in Ebola-related tweets and Internet searches the following day.

Significant Granger-causality provides evidence that X may in fact cause Y (although the caveat that correlation does not equal causation applies). Note, however, that in the case that X does indeed cause Y, the time lag between cause and effect must not be too small relative to the time step of the time series in order for the effect to be detectable with the Granger-causality test.

Contagion Model

To assess the potential contagion of Ebola-related news media in inspiring Ebola-related Google searches or tweets, we employ a Susceptible, Infected, Recovered, Susceptible, and Vector (SIRSV) compartmental model. News videos, V, can "infect" susceptible individuals with interest in the topic, that is subsequently evidenced by Internet searches or tweets. After an average period of $1/\gamma$ days, the individual performs an Internet search or tweet related to the topic, and then "recovers", subsequently becoming less interested with the topic and no longer feeling a need to perform another Internet search or tweet. The model includes the potential that recovered individuals can flow back into the susceptible compartment (i.e. after a period of time the individual may once again become susceptible to being infected with interest in the topic).

The equations of the model are

$$\begin{aligned} \frac{dS}{dt} &= -\beta VS/N - \mu SI/N + \alpha R \\ \frac{dI}{dt} &= +\beta VS/N + \mu SI/N - \gamma I \\ \frac{dR}{dt} &= +\gamma I - \alpha R, \end{aligned} \tag{1}$$

where $N = S + I + R$ is the U. S. population, and $V(t)$ is the temporal evolution of the number of news videos per day. The parameter β is the number of Internet searches or tweets per unit time inspired by one news video shown to a completely susceptible population. The parameter μ represents the rate of searches inspired by things other than news media, such as people talking among themselves about a particular subject. The parameter α represents the rate at which recovered individuals flow back in to the susceptible compartment.

Our model is somewhat similar to the model used in Reference [26] to simulate the spread of ideas within a population, with the difference that we include an exogenous infection component, V, rather than assuming that contagion is only due to intrinsic infectious process within the closed population.

The system of equations in Eq 1 is numerically solved to obtain the estimated number Internet searches or tweets made per unit time, $\gamma I(t)$. The number of news videos varies quite dramatically by day, making numerical solutions unreliable due to stiffness of the model equations if the raw data for the daily number of news videos, $V(t)$, are used; we thus first fit a cubic spline to the daily news video data in order to smoothly interpolate the data in finer time steps.

To estimate the model parameters that optimally describe each data sample, we use the Monte Carlo method for solution of inverse problems to randomly sample the parameters of the model from broad uniform distributions, and calculate the Pearson χ^2 goodness-of-fit statistic comparing the model to the data sample [27, 28]. The uniform distribution sampling range is chosen to be large enough to ultimately include the parameter optimal value and at least a ±5 standard deviation range about that value. In order to determine the parameter hypotheses that minimize the Pearson χ^2 goodness-of-fit statistic, this procedure is repeated at least one million times for each sample with the use of National Science Foundation Extreme Science and Engineering Discovery Environment (NSF XSEDE) high-throughput computing resources (see www.xsede.org, accessed February 9, 2015). To determine the parameter 95% confidence intervals, the Pearson χ^2 statistic is corrected for over-dispersion using the ansatz of McCullagh and Nelder (1989) [29].

If the parameter optimization procedure determines that μ and α are statistically consistent with zero (i.e. that there is no evidence of contagion within the population due to dynamics other than exposure to news videos, and immunity upon recovery is permanent), a more

appropriate model is

$$\begin{aligned} \frac{dS}{dt} &= -\beta VS/N \\ \frac{dI}{dt} &= +\beta VS/N - \gamma I \\ \frac{dR}{dt} &= +\gamma I. \end{aligned} \tag{2}$$

In addition, if the parameter optimization procedure determines that μ, α, and $1/\gamma$ are statistically consistent with zero, the most appropriate model is

$$\begin{aligned} \frac{dS}{dt} &= -\beta VS/N \\ \frac{dR}{dt} &= +\beta VS/N, \end{aligned} \tag{3}$$

where the move to the recovered class happens essentially immediately after exposure to a news video. Given that our data are aggregated by day, "immediately" in this case essentially implies a time frame smaller than a day.

Note that initially the entire population is not typically susceptible to ideation to do an Ebola-related Internet search or tweet due to exposure to an Ebola-related news story aired by Fox News or MSNBC; the people considered "susceptible" in this analysis are people who watch those networks, have access to a computer, and are also susceptible to various levels of ideated panic, ranging from simple interest or curiosity, to actually wondering if they themselves might have Ebola. In the parameter estimation procedure, we thus also estimate the initial fraction of susceptible people in the population for each kind of Ebola-related tweet or Internet search.

As a validation cross-check of the model, we optimize the model parameters to the first half of the time series, and use the resulting model to predict the temporal patterns of the remaining half. We also perform this model validation procedure for a simple linear regression fit, where the time series of the number of Internet searches and tweets are regressed on V. Unlike the mathematical model, the linear regression model does not include the effect of eventual boredom with a topic.

As an additional cross-check of the analysis, we swap I and V in the model, and examine the potential that Ebola-related Internet searches or tweets inspire the temporal patterns in news media coverage, rather than the other way around.

Results and Discussion

For all data samples considered in this analysis, we find that the contagion model parameter μ is statistically consistent with zero (i.e. we find that there is no statistically significant evidence of contagion due to effects other than news videos). We also find that the parameter α is statistically consistent with zero for all samples (i.e. there is no statistically significant evidence that people return to the susceptible class after recovery). In addition, for the Twitter data samples, $1/\gamma$ is statistically consistent with zero, indicating that after viewing a news video the movement to the recovered compartment after doing an Ebola-related tweet occurs within a day. For these data samples, we thus fit the reduced contagion model of Eq 3.

In Table 1, we show the best-fit parameters of the contagion model fit to the data samples. In Table 2, we show the percentage of the variance, R^2, in the data samples described by the best-fit contagion model. We also show the results of the model validation procedure, showing

Table 1. Parameters of the Ebola-related news media contagion model of Eq 2 or Eq 3 (as appropriate to the sample), fit to the Ebola-related Google searches and tweets.

	N	f	$\beta * f$	$1/\gamma$ (days)
Tweets: keyword "Ebola"	251,000	0.0012 [0.0011, 0.0032]	180 [140, 210]	-
Tweets: keywords "Ebola" and "symptoms"	2,350	1e–05 [9.7e–06,2.8e–05]	1.7 [1.3,2.2]	-
Google searches: "Ebola"	26,100,000	0.12 [0.097,0.18]	22000 [17000, 31000]	0.7 [0.3,3.1]
Google searches: "Ebola symptoms"	4,240,000	0.017 [0.015,0.022]	4000 [3200, 5100]	0.7 [0.3,1.9]
Google searches: "Do I have Ebola"	22,200	8.8e–05 [7.6e–05,0.00014]	25 [16, 71]	3.8 [1.4,14.3]

The parameter f is the initial fraction of the population susceptible to news media induced Ebola interest or panic (as manifested by the particular Ebola-related Internet searches or tweets in our samples). The parameter β is the transmission rate, and $1/\gamma$ is the average time, in days between an individual viewing an Ebola-related news video, and performing an Ebola-related Google search or tweet. The average number of particular Internet searches or tweets in our samples inspired by a single news video in the initial susceptible population is $f\beta$. The numbers in the square brackets represent the 95% confidence intervals.

doi:10.1371/journal.pone.0129179.t001

Table 2. The percentage of the variance, R^2, of the Ebola-related Twitter and Google search samples described by the contagion model of Eq 2 or Eq 3 (as appropriate to the sample); shown are the R^2 of the model fit to the full sample, the first half of the sample (model validation training sample), and the extrapolated model prediction for the remaining half of the sample (model validation test sample).

	Contagion Model			Regression Model		
	R^2 full	R^2 train	R^2 test	R^2 full	R^2 train	R^2 test
Tweets: keyword "Ebola"	0.75	0.84	0.63	0.72	0.78	0.44
Tweets: keywords "Ebola" and "symptoms"	0.65	0.81	0.51	0.66	0.77	0.47
Google searches: "Ebola"	0.73	0.79	0.66	0.70	0.76	0.62
Google searches: "Ebola symptoms"	0.76	0.81	0.74	0.62	0.75	0.50
Google searches: "Do I have Ebola"	0.75	0.89	0.37	0.49	0.67	0.15

Also shown are the R^2 for the statistical model, which linearly regresses the data samples on the daily number of Ebola-related news videos.

doi:10.1371/journal.pone.0129179.t002

the R^2 of the model fit to the first half of each data sample, and the R^2 of the subsequent model prediction extrapolated to the last half of the sample. The R^2 of a simple statistical model, linearly regressing the data samples on the daily number of Ebola-related news videos, is also shown. In all cases, the contagion model has better predictive power than the linear regression model.

As a cross-check, we also use the contagion model to examine the possibility that the Ebola-related Internet searches or tweets inspire news videos on the topic. The results are summarized in Table 3. In all cases, we find that the R^2 is much worse for this model compared to the model where we assume that the news videos inspire the Internet searches and tweets.

Also shown in Table 3 are the p-values of the Granger causality test that tests the hypothesis that temporal patterns in news videos Granger-cause temporal patterns in the Internet searches and tweets on the next day, and vice versa. In all cases, there is no statistically significant evidence that Ebola-related Internet searches and tweets Granger-cause temporal patterns in Ebola-related news videos, but there is evidence in several cases that the reverse is true.

Fig 3 shows the best-fit contagion and linear regression models overlaid on the data.

Table 3. The percentage of the variance, R^2, of the data samples described by the contagion model of Eq 1, assuming that the news videos, *V*, cause the patterns seen in the data ($V \rightarrow I$).

	Contagion Model		Granger Causality test	
	R^2 $V \rightarrow I$	R^2 $I \rightarrow V$	p-value $V \rightarrow I$	p-value $I \rightarrow V$
Tweets: keyword "Ebola"	0.75	0.38	0.11	0.89
Tweets: keywords "Ebola" and "symptoms"	0.65	0.41	0.02	0.93
Google searches: "Ebola"	0.73	0.35	0.02	0.37
Google searches: "Ebola symptoms"	0.76	0.34	0.02	0.31
Google searches: "Do I have Ebola"	0.75	0.32	0.09	0.65

Also shown are the R^2 under the assumption that the temporal patterns in the data samples cause the temporal patterns in the news videos ($I \rightarrow V$). The p-values testing for Granger causality between the various time series are also shown.

doi:10.1371/journal.pone.0129179.t003

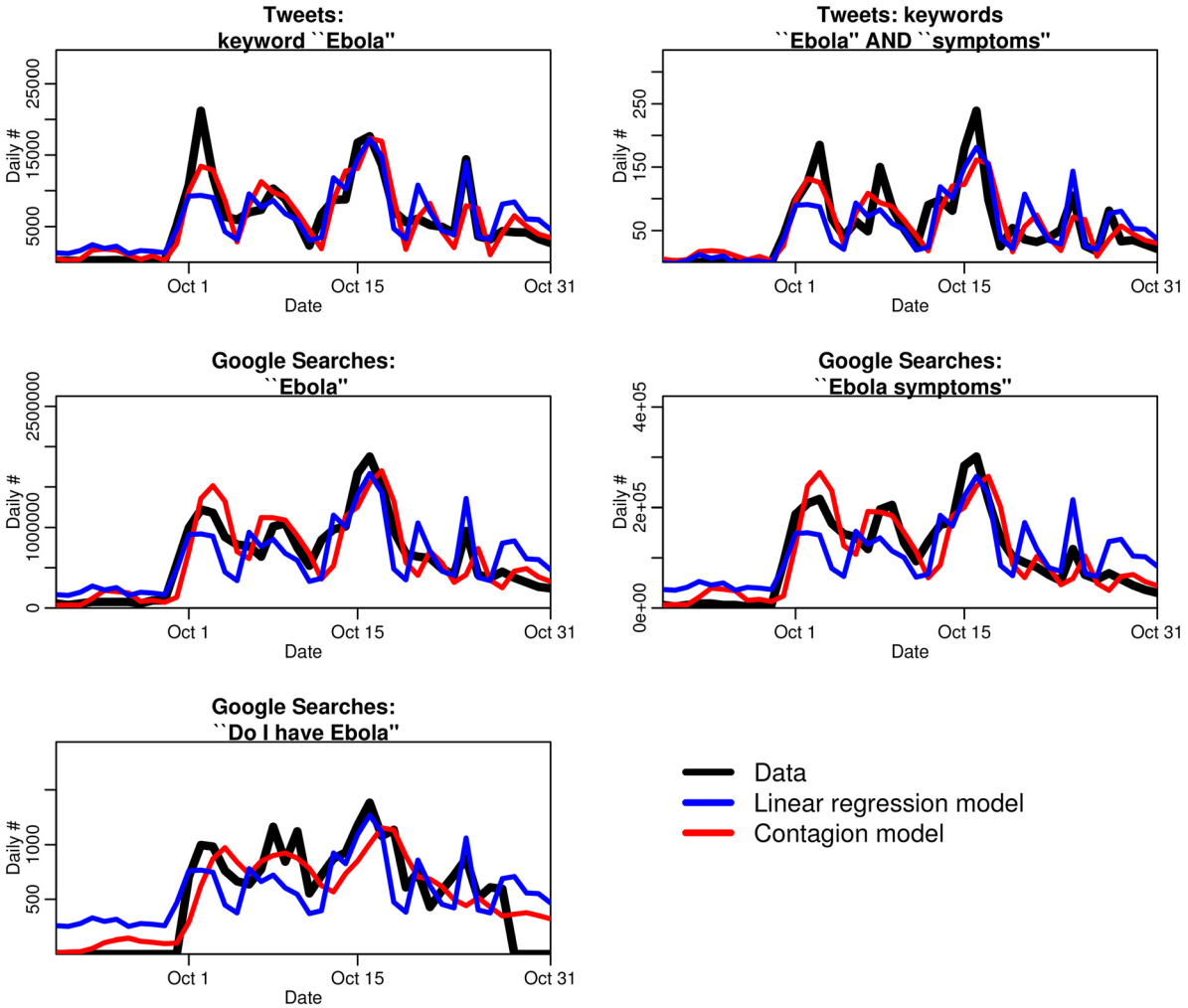


Fig 3. Fits of the news media contagion model, and a simple linear regression model, to the sources of data used in this study. The fits of the linear regression model (shown in blue) tend to be generally too low in the beginning and too high at the end. In contrast, the contagion model (red line) accounts for the boredom effect, where people become more and more disinclined to perform Ebola-related searches or tweets after an extended period of exposure to Ebola-related news-coverage. Incorporation of this dynamic in the model yields significantly better fits to the data compared to the regression model.

doi:10.1371/journal.pone.0129179.g003

Summary

As shown in Table 2, we find that a large fraction of the variance in the data is described by the contagion model (R^2 at least 65% in all cases), and the model validation procedure shows good predictive capability for the extrapolated model to the test samples. The performance of a simple linear regression model is not as good; for all but one sample, the linear regression model yields a lower R^2 than the contagion model, and the model validation procedure reveals that the regression model has generally much poorer predictive capabilities compared to the contagion model. It is thus apparent that in modeling the dynamics of ideation due to exposure to news media, "recovery" and ultimate immunity to further ideation should be taken into account.

As shown in the best-fit model parameters Table 1, we find that a relatively large percentage of the population (up to 18%) is susceptible to ideation with what likely is simple curiosity prompting a Google search for "Ebola", and we find that one Ebola-related news video aired by Fox News or MSNBC on average inspires tens of thousands of such searches in the American population. In contrast, a much smaller percentage of the population appears to be susceptible to ideation to perform more specific searches indicating personal concern about actually having the disease, such as Google searches for "do I have Ebola".

The contagion model we employ assumes that people recover once performing an Ebola-related Internet search or tweeting Ebola-related information, and do not feel the need to seek or disseminate information again. Through comparison of the fit of a contagion model to the fit of a simple linear regression model that does not include the dynamics of the recovery process, we find evidence that indeed recovery and immunity to further ideation does play a role in the overall dynamics of peoples' information seeking behavior. This is concordance with the conclusions of a previous study which examined the information seeking behavior of people during the 2009 A/H1N1 pandemic [30], and determined that information seeking became less common as the pandemic progressed.

Our model assumes that people perform just a single tweet or Internet search before moving to the "recovered" class with permanent immunity. In reality, this simplifying assumption may be violated in some cases, with some people performing several related Internet searches or tweets in a short time period. For instance, the 250,723 tweets related to Ebola used in this analysis were produced by 118,705 unique Twitter users. However, the average time between the first tweet and becoming disinterested with the topic and never tweeting again was 3 days.

In our analysis, we did not examine the sentiment expressed in the Twitter data. Previous studies have shown that negative sentiments tend to be more infectious on social networks than positive sentiments [31]; while we found no evidence of contagion of Ebola-related sentiments within the Twitter community itself in this particular analysis, it may be that sentiment, in addition to temporal patterns, may be used in a real outbreak situation to disentangle the effect of news media contagion, contagion within the social media platform, and effects due to the spread of the disease.

The vast majority of published digital epidemiology results show a positive correlation between digital data and the temporal evolution of epidemics or outbreaks; but what typically are not seen are the analyses that show no significant correlation, likely due to the "file drawer" effect where uninteresting or null results simply are not published [32, 33]. Indeed, several of the authors (ST, DE, SA) can attest to the fact that use of Twitter data to predict outbreaks is fraught with difficulties (unpublished data), and accounting for potential sources of bias is extraordinarily difficult. However, in this new age of readily accessible, and rapidly evolving, temporal and geo-spatial information in social media, digital epidemiology has a hopeful future as a tool to detect newly emerging infectious diseases, and track the spread of established diseases.

The methodology is being continually refined to reduce both Type I and Type II errors (i.e. false positives and false negatives), and efforts are being made to understand the potential biases of the methods. With our analysis, we have explored a source of major potential bias when applying digital epidemiological methodology to emerging disease outbreaks; while media-induced panic is certainly not the only source of bias in such situations, we hope the results of our study will be informative for future analyses.

Supporting Information

S1 Data. The file S1_Data.csv is a file containing the data used in this study.
(CSV)

Acknowledgments

The authors are grateful to Jonathan Dushoff for useful discussions related to this work. The authors also wish to thank the National Science Foundation Extreme Science and Engineering Discovery Environment (NSF XSEDE) high-performance computing initiative for education allocation #DMS140043, which made possible the parameter optimization portion of this analysis. This work was funded in part by the U. S. Department of Homeland Security (DHS) VACCINE Center award #2009-ST-061-CI0001-06, and was also made possible by grant #1R01GM100471-01 from the National Institute of General Medical Sciences (NIGMS) at the National Institutes of Health. This research was also partially supported by the Western Alliance to Expand Student Opportunities (WAESO) Louis Stokes Alliance for Minority Participation (LSAMP) Bridge to the Doctorate (BD) National Science Foundation (NSF) "Multidisciplinary STEM Solutions LSAMP Bridge to the Doctorate" Grant #HRD-1401190, and the Offices of the President and Provost of Arizona State University. The contents of this manuscript are solely the responsibility of the authors and do not necessarily represent the official views of the funding agencies. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Author Contributions

Conceived and designed the experiments: ST CCC. Performed the experiments: ST GB NB SB BE JJ JJG MK ML RM VMM FN KO MLR CR. Analyzed the data: ST GB NB SB BE JJ JJG MK ML RM VMM FN KO MLR CR. Contributed reagents/materials/analysis tools: ST DE SA CCC JM. Wrote the paper: ST SA GB NB SB BE JJ JJG MK ML RM VMM FN KO MLR CR.

References

1. Eysenbach G (2006) Infodemiology: tracking flu-related searches on the web for syndromic surveillance. In: AMIA Annual Symposium Proceedings. American Medical Informatics Association, volume 2006, p. 244.
2. Polgreen PM, Chen Y, Pennock DM, Nelson FD, Weinstein RA (2008) Using Internet searches for influenza surveillance. Clinical infectious diseases 47: 1443–1448. doi: 10.1086/593098 PMID: 18954267
3. Ginsberg J, Mohebbi MH, Patel RS, Brammer L, Smolinski MS, Brilliant L (2008) Detecting influenza epidemics using search engine query data. Nature 457: 1012–1014. doi: 10.1038/nature07634
4. Signorini A, Segre AM, Polgreen PM (2011) The use of Twitter to track levels of disease activity and public concern in the US during the influenza A H1N1 pandemic. PloS one 6: e19467. doi: 10.1371/journal.pone.0019467 PMID: 21573238
5. Brownstein JS, Freifeld CC, Madoff LC (2009) Digital disease detection: harnessing the Web for public health surveillance. New England Journal of Medicine 360: 2153–2157. doi: 10.1056/NEJMp0900702 PMID: 19423867

6. Salathe M, Bengtsson L, Bodnar TJ, Brewer DD, Brownstein JS, Buckee C, et al. (2012) Digital epidemiology. PLoS computational biology 8: e1002616. doi: 10.1371/journal.pcbi.1002616 PMID: 22844241
7. Schmidt CW (2012) Trending now: using social media to predict and track disease outbreaks. Environ Health Perspect 120: a30–a33. doi: 10.1289/ehp.120-a30 PMID: 22214548
8. Eysenbach G (2009) Infodemiology and infoveillance: framework for an emerging set of public health informatics methods to analyze search, communication and publication behavior on the Internet. Journal of medical Internet research 11. doi: 10.2196/jmir.1157 PMID: 19329408
9. Chunara R, Andrews JR, Brownstein JS (2012) Social and news media enable estimation of epidemiological patterns early in the 2010 Haitian cholera outbreak. The American Journal of Tropical Medicine and Hygiene 86: 39–45. doi: 10.4269/ajtmh.2012.11-0597 PMID: 22232449
10. Salathé M, Khandelwal S (2011) Assessing vaccination sentiments with online social media: implications for infectious disease dynamics and control. PLoS computational biology 7: e1002199. doi: 10.1371/journal.pcbi.1002199 PMID: 22022249
11. Hay SI, George DB, Moyes CL, Brownstein JS (2013) Big data opportunities for global infectious disease surveillance. PLoS medicine 10: e1001413. doi: 10.1371/journal.pmed.1001413 PMID: 23565065
12. Balmaseda A, Standish K, Mercado JC, Matute JC, Tellez Y, Saborio S, et al. (2010) Trends in patterns of dengue transmission over 4 years in a pediatric cohort study in Nicaragua. Journal of Infectious Diseases 201: 5–14. doi: 10.1086/648592 PMID: 19929380
13. Chan EH, Sahai V, Conrad C, Brownstein JS (2011) Using web search query data to monitor dengue epidemics: a new model for neglected tropical disease surveillance. PLoS neglected tropical diseases 5: e1206. doi: 10.1371/journal.pntd.0001206 PMID: 21647308
14. Wilson K, Brownstein JS (2009) Early detection of disease outbreaks using the Internet. Canadian Medical Association Journal 180: 829–831. doi: 10.1503/cmaj.090215 PMID: 19364791
15. Zhou X, Ye J, Feng Y (2011) Tuberculosis surveillance by analyzing Google trends. Biomedical Engineering, IEEE Transactions on 58: 2247–2254. doi: 10.1109/TBME.2011.2132132
16. Goel S, Hofman JM, Lahaie S, Pennock DM, Watts DJ (2010) Predicting consumer behavior with Web search. Proceedings of the National Academy of Sciences 107: 17486–17490. doi: 10.1073/pnas.1005962107
17. Choi H, Varian H (2012) Predicting the present with Google Trends. Economic Record 88: 2–9. doi: 10.1111/j.1475-4932.2012.00809.x
18. Cook S, Conrad C, Fowlkes AL, Mohebbi MH (2011) Assessing Google flu trends performance in the united states during the 2009 influenza virus a (H1N1) pandemic. PloS one 6: e23610. doi: 10.1371/journal.pone.0023610 PMID: 21886802
19. Chew C, Eysenbach G (2010) Pandemics in the age of Twitter: content analysis of tweets during the 2009 H1N1 outbreak. PloS one 5: e14118. doi: 10.1371/journal.pone.0014118 PMID: 21124761
20. Olson DR, Konty KJ, Paladini M, Viboud C, Simonsen L (2013) Reassessing Google Flu Trends data for detection of seasonal and pandemic influenza: a comparative epidemiological study at three geographic scales. PLoS computational biology 9: e1003256. doi: 10.1371/journal.pcbi.1003256 PMID: 24146603
21. Metaxas PT, Mustafaraj E, Gayo-Avello D (2011) How (not) to predict elections. In: Privacy, security, risk and trust (PASSAT), 2011 IEEE third international conference on and 2011 IEEE third international conference on social computing (SocialCom). IEEE, pp. 165–171.
22. Cha M, Benevenuto F, Haddadi H, Gummadi K (2012) The world of connections and information flow in twitter. Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on 42: 991–998. doi: 10.1109/TSMCA.2012.2183359
23. Gallup (2013) TV is Americans' main source of news See also: www.gallup.com/poll/163412/americans-main-source-news.aspx (accessed February 28, 2015).
24. Pew Research Center (2014) Key indicators in media and news See also: www.journalism.org/2014/03/26/state-of-the-news-media-2014-key-indicators-in-media-and-news (accessed February 28, 2015).
25. Granger CW (1969) Investigating causal relations by econometric models and cross-spectral methods. Econometrica: Journal of the Econometric Society: 424–438. doi: 10.2307/1912791
26. Bettencourt L, Cintron-Arias A, Kaiser DI, Castillo-Chavez C (2006) The power of a good idea: Quantitative modeling of the spread of ideas from epidemiological models. Physica A: Statistical Mechanics and its Applications 364: 513–536. doi: 10.1016/j.physa.2005.08.083
27. Cowan G (1998) Statistical Data Analysis. Oxford University Press 1998.

28. Mosegaard K and Tarantola A (1995) Monte Carlo sampling of solutions to inverse problems. Journal of Geophysical Research: Solid Earth (1978–2012) 100 (B7): 12431–12447.

29. McCullagh P, Nelder JA (1989) Generalized linear models. London England Chapman and Hall 1983.

30. Tausczik Y, Faasse K, Pennebaker JW, Petrie KJ (2012) Public anxiety and information seeking following the H1N1 outbreak: blogs, newspaper articles, and Wikipedia visits. Health communication 27: 179–185. doi: 10.1080/10410236.2011.571759 PMID: 21827326

31. Salathé M, Vu DQ, Khandelwal S, Hunter DR (2013) The dynamics of health behavior sentiments on a large online social network. EPJ Data Science 2: 1–12.

32. Rosenthal R (1979) The file drawer problem and tolerance for null results. Psychological bulletin 86: 638. doi: 10.1037/0033-2909.86.3.638

33. Dickersin K (1990) The existence of publication bias and risk factors for its occurrence. Jama 263: 1385–1389. doi: 10.1001/jama.1990.03440100097014 PMID: 2406472

Proactive Spatiotemporal Resource Allocation and Predictive Visual Analytics for Community Policing and Law Enforcement

Abish Malik, Ross Maciejewski, *Member, IEEE*, Sherry Towers, Sean McCullough, and David S. Ebert, *Fellow, IEEE*

Abstract— In this paper, we present a visual analytics approach that provides decision makers with a proactive and predictive environment in order to assist them in making effective resource allocation and deployment decisions. The challenges involved with such predictive analytics processes include end-users' understanding, and the application of the underlying statistical algorithms at the right spatiotemporal granularity levels so that good prediction estimates can be established. In our approach, we provide analysts with a suite of natural scale templates and methods that enable them to focus and drill down to appropriate geospatial and temporal resolution levels. Our forecasting technique is based on the Seasonal Trend decomposition based on Loess (STL) method, which we apply in a spatiotemporal visual analytics context to provide analysts with predicted levels of future activity. We also present a novel kernel density estimation technique we have developed, in which the prediction process is influenced by the spatial correlation of recent incidents at nearby locations. We demonstrate our techniques by applying our methodology to Criminal, Traffic and Civil (CTC) incident datasets.

Index Terms—Visual Analytics, Natural Scales, Seasonal Trend decomposition based on Loess (STL), Law Enforcement

1 Introduction

The increasing availability of digital data provides both opportunities and challenges. The potential of utilizing these data for increasing effectiveness and efficiency of operations and decision making is vast. Harnessing this data with effective tools can transform decision making from reactive to proactive and predictive. However, the volume, variety, and velocity of these data can actually decrease the effectiveness of analysts and decision makers by creating cognitive overload and paralysis by analysis, especially in fast-paced decision making environments.

Many researchers in data visualization and visual analytics [37] have proposed interactive visual analytical techniques to aid analysts in these tasks. Unfortunately, most work in this area has required these casual experts (experts in domains, but not necessarily statistics experts) to carefully choose appropriate parameters from a vast parameter space, select the proper resolution over which to perform their analysis, apply appropriate statistical or machine learning analysis techniques, and/or understand advanced statistical significance testing, while accounting for the different uncertainties in the data and processes.

Moreover, the casual experts are required to adapt their decision making process to the statistical analysis space where they need to choose the appropriate time and space scales that give them meaningful analytical and predictive results. They need to understand the role that data sparsity, different distribution characteristics, data variable co-dependencies, and data variance play in the accuracy and reliability of the analytical and prediction results. In moving to this proactive and predictive environment, scale issues become even more important. Not only does the choice of appropriate scales help guide the users' perception and interpretation of the data attributes, it also facilitates gaining new insight into the dynamics of the analytical tasks [42] and the validity of the analytical product: a spatial resolution level that is too fine may lead to zero data input values with no predictive statistical value; whereas, a scale that is too coarse can overgeneralize the data and introduce variation and noise, reducing the value and specificity of the results. Therefore, it becomes critical for forecasting and analysis to choose statistically meaningful resolution and aggregation scales. Utilizing basic principles from scaling theory [42], and Norman's naturalness and appropriateness principles [26], we can both balance and harness these cognitively meaningful natural human-centered domain scales with meaningful statistical scales.

Therefore, in this paper, we present a visual analytics approach that provides casual experts with a proactive and predictive environment that enables them to utilize their domain expertise while exploring their problem and making decisions and predictions at natural problem scales to increase their effectiveness and efficiency in planning, resource allocation, and deployment. Our visual analytics framework [21, 22] provides interactive exploration of multisource, multivariate spatiotemporal datasets using linked views. The system enables the exploration of historic datasets and examination of trends, behaviors and interactions between the different spatiotemporal data elements. The focus of this paper, however, is to provide a proactive decision making environment where historic datasets are utilized at natural geospatial and temporal scales in order to guide future decisions and resource allocation strategies.

In our predictive visual analytics process, we allow users to interactively select and refine the data categories over which to perform their analyses, explore and apply meaningful geospatial (Sections 4.1-4.3) and temporal (Section 4.4) scales and aggregations, apply the forecasting process over geospace (Section 5), and visualize the forecasting results over their chosen geospatial domain. We utilize a Seasonal Trend decomposition based on Loess (STL) [9] approach (Section 3) that utilizes patterns of historical data and apply it in the geospatial domain to predict future geospatial incidence levels. Moreover, this approach provides domain-driven refinement of analysis and exploration to areas and time of significance (e.g., high crime areas or times).

The contributions of our work include these novel natural spatial and temporal analytical techniques, as well as a novel Dynamic Covariance Kernel Density Estimation method (DCKDE) (Section 4.2.2). These contributions can be applied to a variety of spatiotemporal datasets including distribution and logistics, public safety, public health, and law enforcement. We will utilize data from Criminal, Traffic, and Civil (CTC) incident law enforcement datasets in the examples throughout this paper. However, it should be noted that our technique is versatile and can be adapted for other relevant spatiotemporal datasets that exhibit seasonality.

- *Abish Malik, Sean McCullough and David S. Ebert are with Purdue University. E-mail: amalik|mccullo0|ebertd@purdue.edu.*
- *Ross Maciejewski and Sherry Towers are with the Arizona State University. E-mail: rmacieje|smtowers@asu.edu.*

Manuscript received 31 March 2013; accepted 1 August 2013; posted online 13 October 2013; mailed on 4 October 2013.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

2 Related Work

In recent years, there has been much work done in utilizing historic datasets for informing future actions and decisions of decision makers. Below, we discuss previous work in the field of visual analytics, and, since our chosen example domain and implementation is focused on crime data, we also explore previous work in criminology to provide a breadth of the related research areas.

2.1 Predictive Visual Analytics

There have been several visual analytics systems developed in recent years that support data analysis and exploration processes, and provide extensive data modeling and hypothesis generation tools (e.g., [28, 35]). More recently however, researchers have also started progressing toward creating visual analytics systems that incorporate predictive analytics in them. For example, Wong et al. [43] provide a visual interface and an environment that brings together research from several different domains to predict and assess the impact of climate change on U.S. power-grids. Muhlbacher and Piringer [25] provide a visual analytics framework for building regression models. Monroe et al. [23] utilize user-driven data visualizations that enable researchers to gain insights into large healthcare datasets.

Yue et al. [45] created an artificial intelligence based tool that leverages interactive visualization techniques to leverage data in a predictive analytics processes. Their time series modeling technique includes the use of the Box-Jenkins procedure [27]. Other time series modeling techniques extensively used include the ARMA (Auto Regressive Moving Average) [1] and ARIMA (Auto Regressive Integrated Moving Average) models. A summary of some other methods that involve geospatial modeling can be found in [11, 12]. Maciejewski et al. [20] utilize the seasonal trend decomposition by loess smoothing for generating temporal predictions for modeling spatiotemporal healthcare events. They also use the kernel density estimation technique for creating probability distributions of patient locations for use in healthcare data. Our work builds on these ideas where we utilize historic datasets to provide spatiotemporal forecasts into the future. The focus of our work is to explore the issues of geospatial and temporal scales so that casual experts can adapt their decision making process to the statistical analysis space. As such, we apply a user assisted data analysis approach to drive future decisions that helps prevent decision makers from getting over-burdened, while, at the same time, maximizes the utilization of their domain knowledge and perceptual capabilities.

2.2 Crime Hotspot Policing and Intervention

In recent years, there has been much research done that suggest the benefits of hot spot policing in preventing crime and disorder at these crime hotspots (e.g., [2, 3, 4]). Weisburd et al. [41] examine the effect and impact of crime hot spots policing and their findings suggest little negative effects and backlash among the residents of targeted areas of such policing efforts. Sherman [30] also explores the effects of police crackdowns (sudden increase in police presence in specific regions) among several case studies. He notes that while most of the crackdowns appeared to demonstrate initial deterrent effects, the effects decayed after short periods of time. Our work also enables law enforcement decision makers to identify and target crime hotspots by forecasting high probability crime regions based on historic spatiotemporal trends. Our work also factors in the temporal variations within the signals and, as such, provides dynamic hotspot locations for each predicted day.

Goldkamp and V$\hat{i}$lcic$\tilde{a}$ [15] provide insights into unanticipated negative effects of place-oriented enforcement intervention schemes on other societal aspects. They explored an intensive targeted enforcement strategy that was focused on drug crime and its related community effects and examined the overall side effects on the society. Sherman et al. [31] examine and provide an overview of the different aspects of predatory criminal activity at different spatial granularities and how these factors correlate with different aspects of the society. Bruin et al. [7] provide a toolkit that extracts the different factors from police datasets and creates digital profiles for all offenders. The tool then clusters the individuals against the created profiles by using a distance matrix that is built around different attributes (e.g., crime frequency, criminal history of the offenders).

2.3 Predictive Policing

There has been much work done in criminology to study criminal behaviors in order to develop models that predict various offense incidence levels at different spatial aggregation levels. Brown and Oxford [6] study methods that pertain to predicting the number of breaking and enterings in sub-cities and correlate breaking and enterings with different factors including unemployment rates, alcohol sales and previous incidents of crime. Yu et al. [44] also develop a crime forecasting model by employing different data mining classification techniques. They employ several classification techniques including Nearest Neighbor, Decision Tree and Support Vector Machines. Their experiments are run on two different data grid sizes, the 24-by-20 (approx. one-half mile square) and the 41-by-40 square grid cells (approx. one-quarter mile square). They note that the 24-by-20 grids consistently gave them better results than the 41-by-40 grids, which they attribute to the lack of sufficient information at the coarser resolution. Our technique also allows analysts to conduct their predictive forecasting at different spatial resolutions (e.g., over uniform spatial grids and natural underlying spatial boundaries) and temporal granularity levels (e.g., by day, week, month). Furthermore, our system also allows users to create spatial and temporal templates for use in the prediction process.

Monthly and seasonal cycles and periodic properties of crime are well known among criminologists [17]. Felson and Poulson [14] factor in the time of the day variation in the analysis of crime and provide summary indicators that summarize the hour-of-day variations. They provide guidelines for breaking the day into quartiles based on the median hour of crime. We use their guidelines in our work and provide default data driven time-of-day templates over which to forecast crime. We also utilize these techniques and incorporate the seasonality and periodicity properties of crime in order to provide spatiotemporal forecasts of future crime incidence levels.

3 Time Series Prediction using Seasonal-Trend Decomposition based on Loess (STL)

In order to model time series data, we employ the seasonal-trend decomposition technique based on a locally weighted regression (loess) methodology (STL), where a time series signal is considered to consist of the sum of multiple components of variation. To accomplish this, we first utilize the STL method [9, 16] to desynthesize the time series signal into its various components. An analysis of the underlying time series signal Y for CTC data reveals that a square root power transform stabilizes the variability and yields a more Normal distribution of time series residuals, which is a requirement to appropriately model the time series using STL. We consider the time series signal $\sqrt{Y}$ to consist of the sum of its individual components given by $\sqrt{Y_v} = T_v + S_v + D_v + R_v$, where, for the v-th time step, T_v is the inter-annual component, S_v is the yearly-seasonal component, D_v is the day-of-the-week effect, and R_v is the remainder variation component.

To predict using the STL method, we apply the methodology described in [20], where the fitted values $\hat{Y} = (\hat{y}_1, ..., \hat{y}_n)$ generated using the loess operator in the STL decomposition step are considered to be a linear transformation of the input time series $Y = (y_1, ..., y_n)$. This is given by $\hat{y}_i = \sum_{i=1}^{n} h_{ij} y_j \Rightarrow \hat{Y} = HY$, where H is the operator matrix whose (i, j)-th diagonal elements are given by $h_{i,j}$. In order to predict ahead by n days, we append the operator matrix H obtained from predicting ahead within each linear filter in the STL process with n new rows, and use this to obtain the predicted value. The predicted value for day $n+1$ is thereby given by $\hat{y}_{n+1} = \sum_{i=1}^{n} H_{n+1,i}\, Y_i$.

We use this concept of time series modeling and prediction and extend it into the spatiotemporal domain (see Section 5 for details). We further factor in for the sparcity of data in certain geographical regions, and devise strategies to alleviate problems resulting in prediction in these sparse regions (Section 4).

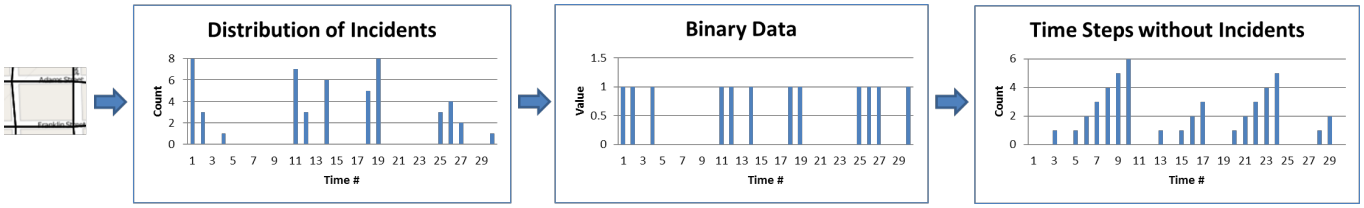


Fig. 1. Our geospatial natural scale template signal generation process. For each geospatial sub-division, the system generates a time series of the number of incidents, converts it into a binary signal, and processes the binary signal to generate the signal used to form the geospatial template.

4 Natural Scale Templates

In order to assist with the analysis process, we provide decision makers with natural scale templates that enable them to focus on appropriate geospatial and temporal resolution levels. These templates enable users to analyze their data at appropriate spatiotemporal granularity levels that help align the scale and frame of reference of the data analysis process with that of the decision making process. These templates also assist users in alleviating the impedance mismatch between data size/complexity and the decision makers' ability to understand and interact with data [29]. We support the creation of both geospatial and temporal templates in our system that facilitate the decision making process. A combination of the generated geospatial and temporal templates provide analysts with an appropriate starting point in the analysis process; thereby, eliminating the need to examine and analyze the entire spatiotemporal parameter space and reducing it to more manageable, appropriate scale levels. To be effective, the design of these scale templates must follow the appropriateness, naturalness, and matching cognitive principles [26]. As Wilkinson and Stevenson both point out [36, 40, 42], simple scaling theory techniques are not sufficient (e.g., axometric scaling theory), but provide useful guidance to primitive scales of reference. The combinations of these design principles and the guidance from these statistical scale papers, provide the motivation and basis for our natural scale templates described below.

4.1 Geospatial Templates

An underlying assumption with using STL to decompose time series is that the data are Normally distributed. The model predictions can get severely biased if this assumption is violated or if data are sparse. To remedy this, we provide methods that help guide users in creating geospatial scales that allow them to drill down to higher incidence regions that may provide better prediction estimates.

4.1.1 Geospatial Natural Scale Templates based on Spatiotemporal Incident Distribution

Our system allows users to narrow down the geographic space for the scope of analysis to regions with higher incidence counts and higher statistical significance for user-selected incident types. Our geospatial natural scale template methodology is shown in Figure 1. In order to generate geospatial templates, the system first fragments the geographic space into either uniform rectangular grids [6] or man-made spatial demarcations (e.g., census blocks). Then, for each subregion, the system generates a time series of the number of incidents that occurred within the subregion over time (e.g., by day, week, month). This signal is further cached for use later in the forecasting process. Next, we convert this time series signal into a binary signal across time, where a 1 represents that an incident occurred on a particular day and a 0 that no incident occurred. We then count the number of 0's between the 1's and progressively sum the number of 0's, outputting the result as another time series signal. As such, this signal is a representation of the number of time steps over which no incidents occurred for the given subregion.

This new time series signal is now utilized in the STL forecasting method (Section 3) and a predicted value is computed for the next day. It should be noted that the resulting time series for regions of lower incidence counts will not be sparse, and consequently, will generate higher predicted values. This process is repeated for all geospatial subregions and a unified picture is obtained for the next day. Finally, we filter out the regions with higher predicted values (low activity) by thresholding for the maximum value. The resulting filtered region forms the initial geospatial template. An example of a created geospatial template using this technique is shown in Figure 4 (Left).

4.1.2 User Refinement of Geospatial Template using Domain Knowledge

The geospatial template provides regions with relatively higher incident rates. The system further allows users to use their domain knowledge and interactively refine these template regions into sub-divisions. For example, users may choose to sub-divide the formed template regions by natural or man-made boundaries (e.g., state roads, rivers, police beats), or by underlying features (e.g., known drug hotspots). The system also allows users to explore the predicted future counts of the created sub-regions by generating an incidence count vs. time signal for each disjoint region and applying our forecasting methodology (Section 3) to find a predicted value for the next day. The results are then shown as a choropleth map to users (e.g., Figure 4 (Right)). These macro-level prediction estimates further assist decision makers in formulating high-level resource allocation strategies.

4.2 Kernel Density Estimation

One of the challenges with using the spatial distribution of incidents in a geospatial predictive analytics process is that it can exacerbate the problem of generating signals with low or no data values. To further refine our prediction model in geospace, we utilize a Kernel Density Estimation (KDE) technique to spread the probability of the occurrence of incidents to its neighboring regions. The rationale behind this is that criminology research has shown evidence that occurrence of certain types of crimes (e.g., residential burglary) at a particular region puts neighboring regions at an elevated risk [13, 18, 32].

Furthermore, crime also tends to be clustered in certain neighborhoods, and the probability of a crime occurring at a particular location can be highly correlated with the number of recent crimes at nearby locations. We incorporate this concept in a novel kernel density estimation method described in Section 4.2.2, where the kernel value at a given location depends on the locations of its k-nearest incidents. In addition, kernel density estimation methods take into account that crimes in low-crime or sparsely populated areas have low incidence, but non-zero probability. We utilize two interchangeable density estimation techniques in our implementation.

4.2.1 Kernel Scale based on Distance to the k-th Nearest Neighbor

To account for regions with variable data counts, we utilize a kernel density estimation technique and use a dynamic kernel bandwidth [33]. We scale the parameter of estimation by the distance from the point x to its kth nearest neighbor X_i. This is shown in Equation 1.

$$\hat{f}(x) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{max(h,d_{i,k})}K\left(\frac{x-X_i}{max(h,d_{i,k})}\right) \quad (1)$$

Here, N is the total number of samples, $d_{i,k}$ the distance from the i-th sample to the k-th nearest neighbor and h is the minimum allowed kernel width. We use the Epanechnikov kernel [33] to reduce calculation time, which is given by $K(\mathbf{u}) = \frac{3}{4}(1-\mathbf{u}^2)1_{(||u||\leq 1)}$. Here, the

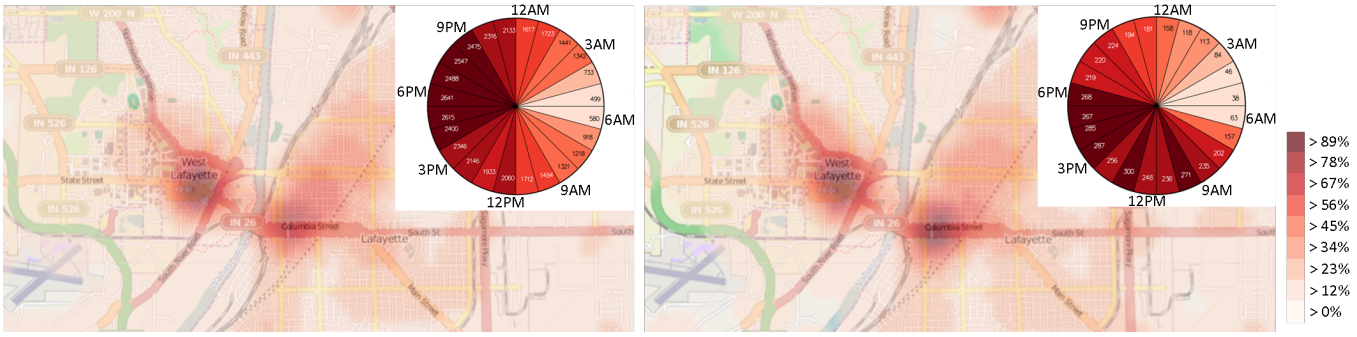


Fig. 2. Spatiotemporal distribution of historical CTC incidents for Tippecanoe County for (Left) 3/11/2012 through 3/10/2014, and (Right) for all Tuesdays in March in the past 10 years.

function $1_{(||u||\leq 1)}$ evaluates to 1 if the inequality is true and to 0 otherwise. In cases where the distance from the i-th sample to the k-th nearest neighbor is 0 (e.g., multiple calls from the same address), we force the variable kernel estimation to a minimum fixed bandwidth h. Making the kernel width placed at the point X_i proportional to $d_{i,k}$ gives regions with sparse data a flatter kernel, and vice-versa.

4.2.2 Dynamic Covariance Kernel Density Estimation Technique (DCKDE)

The kernel in the previous method is based on the distance from an incident location to its k-th nearest neighbor, which provides a flatter kernel for sparse regions. In a new kernel method, we use the information from all k-nearest neighbors to calculate the width of the kernel (rather than the most distant neighbor), thus reducing stochastic variation on the width of the kernel. As such, we fragment the geospatial region into rectangular grids and then utilize a Gaussian kernel at every grid node that is based on the covariance matrix of the location of the center of each node $\mathbf{X} = \{x, y\}$ and its k-nearest neighbor incidents [39]. Therefore, the kernel value is influenced by the k-nearest neighbors and provides a wider kernel in sparsely populated regions that enables the model prediction to be small but non-zero and also takes into account correlations between latitude and longitude; thus, improving the accuracy of the estimates. The value stored at each node location is given by $\delta(\mathbf{X}) = \frac{1}{2\pi|V|} e^{-\frac{1}{2}(\mathbf{X}-\boldsymbol{\mu})^T V^{-1}(\mathbf{X}-\boldsymbol{\mu})}$, where $\boldsymbol{\mu} = \{\mu_x, \mu_y\}$ is the mean along the x and y directions of the k nearest neighbors and their covariance matrix V is defined as:

$$V = \begin{bmatrix} \sigma_x^2 & cov_{x,y} \\ cov_{x,y} & \sigma_y^2 \end{bmatrix} \tag{2}$$

Here, σ_x^2 and σ_y^2 is the variance along the x and y dimension respectively, and $cov_{x,y} = \sum_{i=1}^{k} \frac{(x_i-\mu_x)(y_i-\mu_y)}{k-1}$ is the sample covariance between x and y.

4.3 Neighbors with Similar Spatio-Demographics

For regions that generate a signal of lower statistical significance for the user selected categories, we provide the option to explore data in similar neighborhoods. For each census block, we utilize spatio-demographic census data to find those census blocks that exhibit similar spatial demographics. The rationale behind finding similar neighborhoods lies in the fact that regions with similar demographics tend to exhibit similar trends for certain types of crime [24, 34].

The process of finding similar census blocks for a given census block X includes computing the similarity distance from X to all neighboring census blocks that lie within a d mile radius from the centroid of X. The d mile radius constraint is imposed to factor in for Tobler's first law of geography [38] that suggests that near regions are more related to one another than distant regions. We use $d = 3.0$ miles in our implementation [8]. As such, the similarity distance between two census blocks A and B given k census data variables is given by $S_{A,B} = \sqrt{\sum_{i=1}^{k}(A(V_i) - B(V_i))^2}$, where $A(V_i)$ and $B(V_i)$ are the corresponding census data variable values (e.g., race, income, and age demographic data) for census blocks A and B respectively. Finally, the top N census blocks with the smallest similarity distance values are chosen as the similar census blocks for the given census block X. We use $N = 5$ as a default value in our implementation, but provide users with options to change this value on demand. We note that our future work includes extending this concept of finding similar neighborhoods to determining similar data categories for predictive purposes.

The system now provides users with the ability to generate *similar neighborhood* prediction maps where the prediction for a given census block X depends on the historic time series data of its N similar census blocks in addition to the past data of the census block X itself. Here, the input time series for the census block X used in the prediction algorithm is the per time step average of the N similar census block signals combined with the original signal from census block X. The resulting prediction maps incorporates the influence of incidence rates in neighborhoods that share similar spatio-demographic data.

4.4 Temporal Natural Scale Templates

As noted previously in Section 2.3, crime trends exhibit not only monthly and seasonal trends, but also shows day-of-the-week and hour-of-day variations. The prediction maps produced by the methods described so far provide prediction estimates over 24-hour periods. This information, albeit valuable to the law enforcement community in developing resource allocation strategies for their precincts, provides little detail of the 24-hour distribution of crime. In this section, we describe our approach to assist users in creating temporal scales.

4.4.1 Interactive Clock Display

Figure 2 (Top-Right) shows our interactive clock view that enables a radial display of temporal hourly data. The clock view provides a way for users to filter the data by the hour by interactively clicking on the desired hours, thereby filtering down the data for use in the prediction process. Users may use the clock view display to obtain a visual summary of the hourly distribution of the incidents and consequently make informed decisions on creating temporal templates over which good prediction estimates may be established.

4.4.2 Factoring in for Monthly and Day-of-the-Week Variations

In addition to utilizing the seasonal trend decomposition technique described in Section 3 to decompose the time series signals into its various components, we also utilize a direct approach where we allow users to create their own custom monthly and/or daily templates. Certain crimes tend to peak on certain days of the week (e.g., alcohol related violations tend to be higher over the weekend), whereas other crimes tend to be lower on other days (e.g., reported burglaries drop over the weekend). As such, we factor for these effects directly in the system and allow users to filter data specifically by month and/or by day-of-the-week. This further assists decision makers in developing and refining their resource allocation strategies.

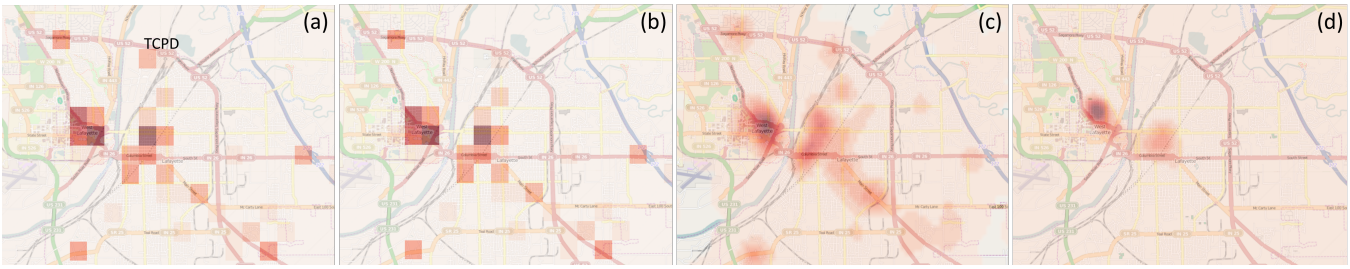


Fig. 3. Geospatial prediction results for 3/11/2014 for Tippecanoe County obtained using our STL forecasting methodology. (a) Predicted choropleth map for rectangular grids of dimension 64×64 using incidence count time series by day. (b) Refined predicted map after removing TCPD location from (a). (c) Predicted map using KDE based on the distance to the k-th nearest neighbor approach (Section 4.2.1). (d) Forecast map using DCKDE method (Section 4.2.2).

4.4.3 Refinement using Summary Indicators

We extend the method described in [14] to further assist users with refining and choosing appropriate hourly templates in the prediction process. In this method, the system computes the *median minute* of CTC incident for the selected 24-hour binning period that provides information about when exactly half of the incidents for the selected date range and offense types have occurred. Next, to get an indication of the dispersion of crime within the 24-hour period, the system computes the *first quartile minute* and *third quartile minute* for the selected data, which are the median times of the first and second halves of the 24-hour period from the median minute respectively. Finally, as temporal data can be inaccurate with many incidents that have missing time stamps, we provide users with an accuracy indicator to show the percentage of cases with valid time stamps. These summary indicators, along with the temporal templates described above, enable users to further refine their selected temporal templates for use in the prediction process. Example scenarios where these indicators are used are provided in Section 6.

5 Geospatial Prediction

The described visual analytics process involves a domain expert selecting appropriate data parameters, applying desired data filters and generating spatial and temporal natural scale templates using the methods described in Section 4. Next, the system incorporates the STL forecasting method (Section 3) and extends it to the geospatial domain to provide prediction estimates for the next N time steps (e.g., days, weeks, months). We now list the steps involved in our geospatial prediction methodology:

1. **Dividing geospace into sub-regions**: The first step in our methodology, just like in Section 4.1.1, involves subdividing geospace into either uniform rectangular grids of user specified resolutions or man-made geospatial boundaries.
2. **Generating the time series signal**: The system then extracts a time series signal for each sub-division. We allow two types of signals to be extracted for each sub-division: (a) incidence count vs. time step, and (b) kernel value vs. time step. Note that the signal generated in (a) is the same as that produced in Section 4.1.1 (Figure 1 (Distribution of Incidents)). The kernel values used in (b) are generated using any one of the methods described in Section 4.2.
3. **Forecasting**: The time series signal generated for each spatial unit is then fed through the STL process described in Section 3 where a forecast is generated for the next N time steps (e.g., days, weeks). This process is repeated for all region sub-divisions and prediction maps are finally obtained for the next N time steps.
4. **Visualizing results**: Finally, the results of our forecasting method are provided to the user either in the form of a choropleth map or a heatmap.

When users choose to fragment the geospace into uniform rectangular grids, we provide them with the ability to select the resolution level, or, in other words, the grid size of each grid. An incidence count vs. time step signal is then generated for each sub-region. It is important to note here that a grid resolution that is too fine may result in a zero count vs. time step signal that has no predictive statistical value. On the other hand, a grid resolution that is too coarse may introduce variance and noise in the input signal, thereby over-generalizing the data. An evaluation of our forecasting approach (Section 7) indicates that an average input size of 10 samples per time step provide enough samples for which our method behaves within the constraints and assumptions of our STL forecasting approach. We utilize this metric in our system in order to determine the applicability of our forecasting method for a particular sub-region.

Figure 3 shows a series of examples that demonstrate our geospatial prediction results using the methods described in this section. Here, the user has selected all CTC incidents for Tippecanoe County, IN, and is using 10 years' worth of historical data (3/11/2004 through 3/10/2014) to generate forecast maps for the next day (i.e., for 3/11/2014). Figure 3 (a) shows the prediction results when Tippecanoe County, IN is fragmented into rectangular grids of dimension 64×64. The input data for each sub-region consists of daily incidence count data over the last 10 years. This method, unlike the KDE methods, does not spread the probability to surrounding neighborhood regions when an incident occurs at a particular place. As a result, this method treats each region independently, and can be used when there are no correlations between geospatial regions (e.g., commercial vs. residential neighborhoods). This method can also be useful in detecting anomalous regions and regions of high predicted levels of activity. For example, the user notices something peculiar from the results in Figure 3 (a): a predicted hotspot occurs prominently over the Sheriff's office and county jail location (labeled as TCPD in Figure 3 (a)). This occurs because the default geospatial location of many incidents are logged in as the county jail, especially when arrests are associated with cases. To remedy for this, the user can refine the underlying geospatial template (Section 4.1.2) and dynamically remove this location from the geospatial template. The refined prediction map generated is shown in Figure 3 (b).

Figures 3 (c and d) show the predicted results of using the kernel density estimation based on the distance to the k-nearest neighbor approach (Section 4.2.1) and the DCKDE technique (Section 4.2.2), respectively. The KDE method applied to generate the prediction map in Figure 3 (c) provides a flatter kernel for relatively low-crime regions. As a result, the prediction map provides lower, but non-zero, predictions for these regions. The kernel width computed using this method is based on the distance from a point x to its kth nearest neighbor only. The DCKDE method, on the other hand, assumes that the probability of the occurrence of an incident at a particular location is correlated with the number of recent incidents at nearby locations. Accordingly, this method utilizes information from *all* k-nearest neighbors in calculating the kernel value. Thus, the regions with persistently higher incident concentrations generate focused hotspots when forecasting is performed using the DCKDE method. Finally, it should be noted that each method provides users with different insights into the dynamics of the underlying processes, and users can use their domain knowledge to further refine the results to make informed decisions.

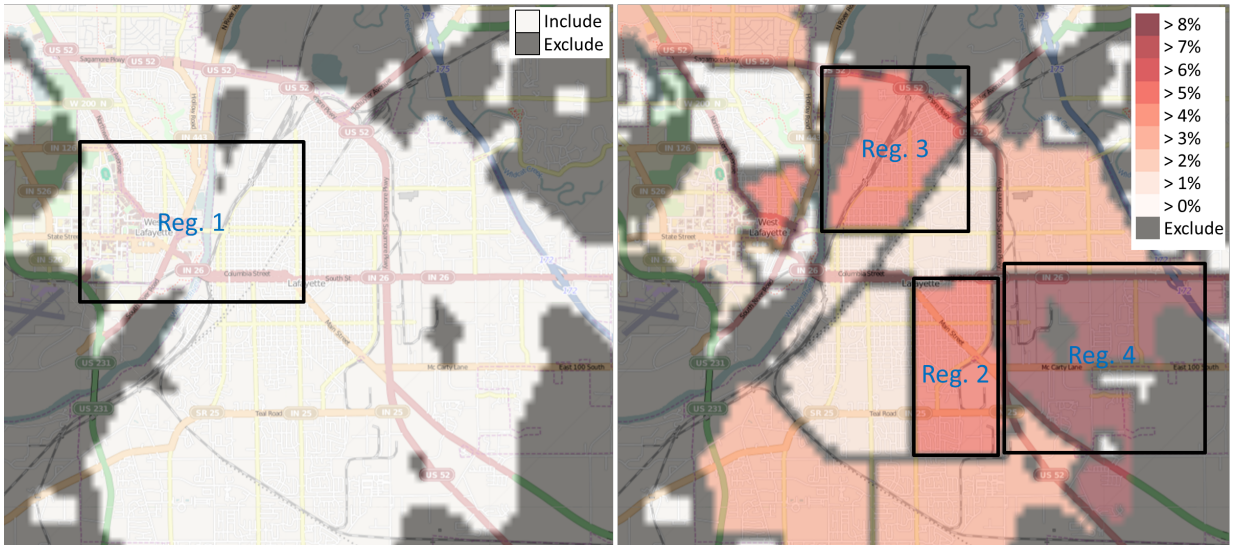


Fig. 4. (Left) Geospatial template generated for Tippecanoe County using 10 years' worth of historical data. (Right) Choropleth map showing the distribution of predicted incidents for 3/11/2014 by police beats for Tippecanoe County. Users may further select regions on the map (e.g., Reg. 1-4) to generate detailed predictions for the selected regions (Figure 5).

6 Case Study: Forecasting Future Criminal, Traffic and Civil (CTC) Incidence Levels

In this section, we demonstrate our work by applying our spatiotemporal natural scale template methodology to forecast for CTC incidence levels in Tippecanoe County, IN, U.S.A. This dataset consists of historical reports and provides several different attributes, including the geographic location, offense type, agency, date, and time of the incident. This dataset contains an average of 31,000 incidents per year for Tippecanoe County, and includes incidence reports for different categories of CTC incidents (e.g., crimes against person, crimes against property, traffic accidents). We use 10 years worth of historical data for this analysis. We provide a workflow when using our system in the analysis process.

Forecasting for all geospatial CTC incidents

Here, we describe a hypothetical scenario in which a law enforcement shift supervisor is using our system to develop resource allocation strategies for Tippecanoe County over the next 24 hour period for Tuesday, March 11, 2014. The supervisor is interested in developing a high-level resource allocation strategy, in particular, by police beats for the next 24 hour period. Law enforcement officers are generally assigned to a particular law beat and patrol their beat during their shift hours when not responding to a call for service. The supervisor is also interested in determining which hotspot locations to focus on for larger police beats. Finally, he also wants to refine the developed resource allocation strategy to factor in for the hourly variation of crime. To develop an appropriate resource allocation strategy, the shift supervisor performs several different analyses that are described in the following subsections. Although our example uses data for all CTC categories as inputs, users may filter their data using any combinations of CTC categories (e.g., crimes against property, person) to further refine their resource allocation strategy.

Overall daily resource allocation

The shift supervisor begins his process by visually exploring the spatiotemporal distribution of historical incidents using our system. When working through the system, the supervisor then visualizes the geospatial and hourly distribution of the incidents that occurred over the past 2 years, as shown in Figure 2 (Left). The supervisor notes several hotspots emerge for the selected period. The locations of these hotspots match with his domain knowledge of the area (e.g., city downtown regions, shopping center locations across town). The static image of the aggregate data, however, does not factor in the inherent spatiotemporal data variations, and basing a resource allocation decision on this image alone would be insufficient. The supervisor is also aware of the fact that police presence can act as a deterrent for certain types of crimes, and, therefore, wants to diversify and maximize police presence in these hotspot areas.

Next, the supervisor wants to factor for monthly and day-of-the-week patterns in his analysis. As such, he visualizes the geospatial and hourly distribution of all CTC incidents that occurred on any Tuesday in the month of March over the past 10 years (Section 4.4.2). The result is shown in Figure 2 (Right). The supervisor notes a slightly different geospatial distribution emerges as a result, with the intensity of hotspots shifting towards the east downtown Lafayette region. In this case, it also becomes apparent that for the 24-hour distribution, 10 AM, 1 PM and 3 PM-6 PM emerge as high activity hours.

Allocating resources by police beats

In order to narrow down the geospace and focus on relevant geographic locations, the supervisor decides to apply our geospatial template generation technique (Section 4.1) with all CTC incidents selected using 10 years' worth of historical data (i.e., from 3/11/2004 through 3/10/2014). The resulting geospace generated is shown in white in Figure 4 (Left). The supervisor notes that the resulting regions correspond to highly populated areas, and exclude areas of infrequent occurrences. Next, the system provides a total predicted number of incidents, N, for March 11, 2014 for the filtered geospatial region. This is done by generating a total incidence count vs. day time series signal using the past 10 years' worth of data and applying the STL forecasting method described in Section 3. Here, N is 59 incidents.

Next, the supervisor is interested in obtaining a high level overview of the distribution of the predicted incidents over geospace, and, in particular, by police patrol routes. As such, the supervisor uses our system and fragments the generated geospatial template using the city law beats shapefile. The resulting geospace is shown in Figure 4 (Right). In order to distribute the total predicted 59 incidents across police beats, the system computes an incidence count vs. day time series signal for each disjoint geospatial region and computes the predicted number of incidents n_i for each region (Section 3). Next, the probability of an incident within each disjoint region is calculated using the formula $p_i = n_i/N * 100$. The results of this operation are then shown to the user as a choropleth map, where each disjoint region is colored according to its value on a sequential color scale [5] (Figure 4 (Right)).

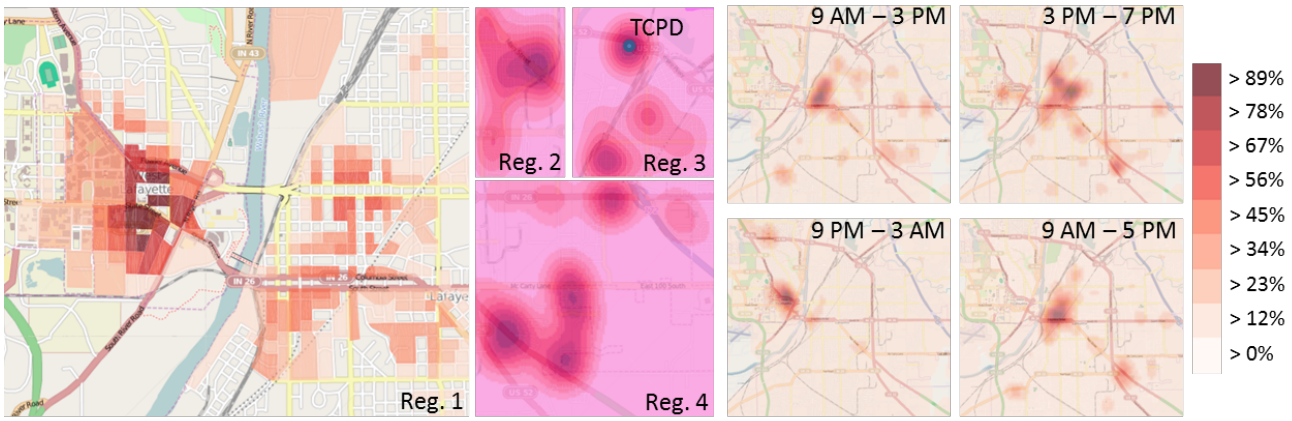


Fig. 5. User refinement of geospatial resource allocation strategy. The user has chosen to visualize predicted hotspots for regions labeled in Figure 4 (Regions 1 through 4), and for Tippecanoe County over hourly temporal templates.

Geospatial resource allocation strategy refinement using domain knowledge

While the high level police beat prediction map (Figure 4 (Right)) suggests putting a heavier emphasis on the eastern police beats of the city, the prediction results in Figure 3 indicate a more localized concentration of incidents at the city downtown locations. The shift supervisor may use these results and allocate higher resources to the eastern police beat of the city (Reg. 4 in Figure 4), and allocate a smaller number of resources, but at more concentrated locations in the downtown (Reg. 1 in Figure 4).

Now, the supervisor is interested in further refining her geospatial resource allocation strategy. First, she turns to the predicted hotspot regions in the city downtown regions (Reg. 1 in Figure 4). She decides to utilize the census blocks spatial boundary information and divides the geospace into census blocks. Next, she uses the method described in Section 5 to create a predicted choropleth map based on census blocks for the region. The result of this operation is shown in Figure 5 (Reg. 1). Here, the supervisor has chosen to use the kernel values obtained from the method described in Section 4.2.1 and spread them across the underlying census blocks for generating these results.

To obtain detailed predictions for the eastern city police beat region (Reg. 4 in Figure 4), the shift supervisor uses a different approach where she draws a region around the selected beat using the mouse and restricts the forecast to the selected region. The result of this operation is shown in Figure 5 (Reg. 4). From domain knowledge, she knows that this area has a high concentration of shopping centers. The hotspots obtained in Figure 5 (Reg. 4) align with these locations. Finally, the supervisor generates similar heatmaps for regions labeled as Reg. 2 and 3 in Figure 4, the results of which are shown in Figures 5 (Reg. 2 and 3), respectively. Note that the county jail location is once again a hotspot in Figure 5 (Reg. 3). With these detailed results in hand, the shift supervisor is able to devise an optimal resource allocation strategy for the next 24 hour period in Tippecanoe County.

Applying temporal templates

Finally, in order to refine her resource allocation strategy to different portions of the day, the shift supervisor chooses to apply the summary indicators method (Section 4.4.3). She finds that the first, median, and third quartile minutes for CTC incidents that occurred in the past 10 years were 9:25 AM, 3:11 PM and 7:28 PM respectively. She also notes that these indicators correspond with the hourly distribution of incidents using the clock view display in Figure 2. Therefore, the supervisor chooses two hourly templates using these summary indicators: (a) 9 AM through 3 PM, and (b) 3 PM through 7 PM. The supervisor also creates two other hourly templates: 9 PM through 3 AM to capture night time activity, and 9 AM through 5 PM to capture working hours of the day. She then uses the kernel density estimation method (Section 4.2.1) and re-generates prediction maps for March 11, 2014. These results are shown in Figure 5. As expected, the supervisor notes the shift in hotspot locations through the 24 hour period, which further enables the refinement of the resource allocation strategy for the different portions of the 24 hour period.

7 Model Evaluation and Validation

In order to evaluate our methodology, we conducted a series of statistical tests to understand the behavior and applicability of our approach in the spatiotemporal domain. Our validation strategy involved testing for the empirical rule of statistics, which describes a characteristic property of a Normal distribution: 95% of the data points are within the range $\pm 1.96\ \sigma$ of μ, where μ and σ are the mean and standard deviation of the distribution, respectively [10]. In order to help alleviate the challenges resulting due to the sparseness of the underlying data, we performed our analyses over a weekly data aggregation level. Our approach involved testing whether the 95% prediction confidence interval bound acquired for the geospatial predictions using our forecasting approach holds when compared against observed data [19]. This confidence bound would be violated if the variance of the observed data is higher (i.e., overdispersed data) or lower (i.e., underdispersed data) than that dictated by the prediction confidence bound. When the 95% prediction bounds are met as expected, and the data conforms to the Normal regime, the applicability of our spatiotemporal STL forecasting method is established.

Building on our STL based time series prediction discussion from Section 3, the variance of the fitted values $\hat{Y} = (\hat{y}_1, ..., \hat{y}_n)$ using the loess operator in the STL decomposition step is given by $Var(\hat{Y}_i) = \hat{\sigma}^2 \sum_{j=1}^{n} H_{ij}^2$ [20]. Here, $\hat{\sigma}^2$ is the variance of the input time series signal Y, and is estimated from the remainder term R_v. Subsequently, the variance for the predicted value $\hat{Y}_{n+1}$ for time step $n+1$ is given by $Var(\hat{Y}_{n+1}) = \hat{\sigma}^2(1 + \sum_{j=1}^{n} H_{n+1,j}^2)$. This provides the 95% prediction interval as $CI_{n+1} = \hat{Y}_{n+1} \pm 1.96\sqrt{Var(\hat{Y}_{n+1})}$.

Next, we performed a series of analyses at varied geospatial and temporal scales, and for different data categories. The geospace was first fragmented into sub-regions (either rectangular grids or using man-made boundaries), and time series signals were generated for each geospatial sub-region. In our analyses, we utilized a sliding time window of size 3 years (i.e., 3×52 weeks) that provided enough samples above the Nyquist frequency for the STL forecasting technique. Forecasting was performed using the methods described in Sections 5 and 7.1. We provide our evaluation methodology and results in the subsequent sub-sections.

7.1 Modified STL forecasting method to factor in for weekly data aggregation

As described earlier in Section 3, a time series signal $\sqrt{Y}$ can be considered to consist of the sum of its inter-annual (T_v), yearly-

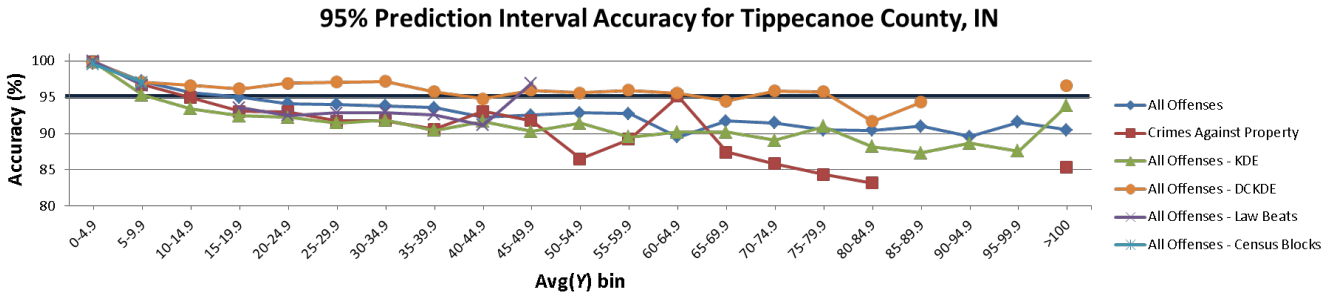


Fig. 6. 95% prediction interval accuracy vs. Avg(Y) for different CTC offenses for Tippecanoe County, IN. Here, geospace has been fragmented into rectangular grids of dimension $k \times k$ ($\forall\, k \in [1,128]$), and by law beats and census blocks.

seasonal (S_v), day-of-the-week (D_v), and remainder variation (R_v) components. However, since we used a weekly aggregation of data, the day-of-the-week component (D_v) must be excluded. Therefore, the time series signal gets modified to $\sqrt{Y_v} = T_v + S_v + R_v$. The prediction step, which involves predicting the value for week $n+1$, remains the same as given in Section 3.

7.2 95% prediction interval accuracy vs. input data average (Avg(Y))

In this method, the geospace was first fragmented into either: (a) rectangular grid regions of dimension $k \times k$ ($\forall\, k \in [1,128]$, with 128 chosen as upper threshold to provide a fine enough geospatial resolution), or (b) man-made geospatial regions (e.g., census blocks, census tracts, law beats, user-specified regions). For each geospatial region, we first generated the incidence count vs. week signal (denote this signal as Y) for a time window of n weeks beginning from the week of, e.g., 1/1/2009. We then used the modified STL forecasting method (Section 7.1) to calculate the 95% prediction interval CI for the predicted week $n+1$, and tested whether the observed data for week $n+1$ fell within the calculated 95% prediction interval for that geospatial region. The average of the input signal Y, $Avg(Y)$, was also calculated.

Next, the input time window was shifted by one week to generate the corresponding incidence count vs. week signal (so, this signal would begin from the week of 1/7/2009). We again computed $Avg(Y)$, and CI for the predicted week $n+1$. As before, we tested whether the observed data for the predicted week $n+1$ fell within the calculated 95% prediction interval. We repeated the process by sliding the time window till it reached the end of available data. For each $Avg(Y)$ value, we maintained two counters that kept track of the number of instances the observed data was within the 95% prediction interval ($C_{Correct}$), and the total instances encountered thus far (C_{Total}). Finally, $Avg(Y)$ values were binned, and $C_{Correct}$ and C_{Total} were summed for each bin. The 95% prediction interval accuracy for each $Avg(Y)$ bin is then given as $\frac{\sum_{bin} C_{Correct}}{\sum_{bin} C_{Total}} \times 100\%$.

7.3 Results and discussion

Figure 6 shows the 95% prediction interval accuracy results for different CTC offenses for Tippecanoe County, IN using the method described in Section 7.2. As can be observed from these results, when the average bin values are low (e.g., less than 10 input samples), the accuracy levels are higher than the expected 95% confidence bound. This indicates that the data are underdispersed for lower input values. In other words, the variance of the observed data is lower than that of the 95% prediction bound when the underlying data are sparse. This conforms to the expected behavior for predicting using our STL forecasting technique: the model predictions get biased if the underlying data are too sparse.

As the input signal average ($Avg(Y)$) values get larger (i.e., more than 10 samples per time step), the prediction accuracy starts to converge at around the expected 95% accuracy level. For example, the prediction interval accuracy for all offenses converges at around 93%. Also, note that the prediction accuracy using the DCKDE method (Section 4.2.2) converges close to the 95% accuracy level; thereby, indicating the efficacy of the technique. It should be noted that since the underlying processes being modeled here (e.g., CTC incidents) are inherently stochastic in nature, perfect 95% confidence bounds will not be achieved (as can be seen from the results in Figure 6). Furthermore, with an uncertain probability distribution of the underlying data, our application of the square root power transform may not guarantee homoscedasticity (i.e., stabilization of variability). This also contributes to our system not achieving perfect 95% confidence bounds. However, even though perfect confidence bounds are not achieved (as can be observed from Figure 6), the accuracy converges close to the 95% bounds. These results show that the underlying data are Normally distributed for higher values of $Avg(Y)$; thereby, satisfying the underlying assumptions of our method used to estimate the 95% confidence interval. This establishes the validity of the claims of our STL prediction methodology in the geospatial domain that the prediction modeling method works as expected as long as the underlying assumptions of the method are satisfied by the data.

Figure 6 shows the 95% prediction interval accuracy vs. input data average results (Section 7.2) for man-made geospatial regions (census blocks and law beats). These results show that the confidence bounds using census blocks are invariably higher than the expected 95% bound, which indicates that the underlying data are underdispersed. Census blocks are small geospatial units, typically bounded by streets or roads (e.g., city block in a city). The smaller $Avg(Y)$ values for census blocks in Tippecanoe County in Figure 6 (less than 10 input samples) further highlight the sparsity of input data. The combination of higher prediction interval accuracy levels and lower $Avg(Y)$ values are telltale for the data sparseness issues we have described, and suggest that the signals generated using census blocks have low predictive statistical power. This further underlines the need to intelligently combine geospatial regions of lower statistical values to obtain a signal of higher predictive power (e.g., as was done in Section 4.3). The 95% prediction interval accuracy results obtained using law beats in Figure 6, on the other hand, shows the accuracy converging at around the expected 95% confidence interval for higher $Avg(Y)$ values (more than 10 input samples). These results provide further evidence that as the underlying data values become larger and begin to conform to the Normal regime, our geospatial prediction methodology provides prediction estimates that are within the expected 95% prediction confidence interval. This further bolsters the applicability and validity of our STL prediction methodology in the geospatial domain.

We also applied the method described in Section 7.2 to all CTC incident category data and generated 95% prediction interval accuracy vs. the input signal average value ($Avg(Y)$) plots for different grid resolutions k. These results are shown in Figure 7. The results indicate that 95% prediction interval accuracy converges at or around the 95% confidence level for large enough $Avg(Y)$ values (i.e., for $Avg(Y)$ bigger than 10). The results indicate that our methodology behaves within the constraints of the Normal regime at higher $Avg(Y)$ values for the different grid dimensions. Also, note that smaller grid dimensions (k) correspond to larger geospatial sub-divisions; and accordingly, smaller k values generate signals of larger counts per bin (i.e., larger $Avg(Y)$

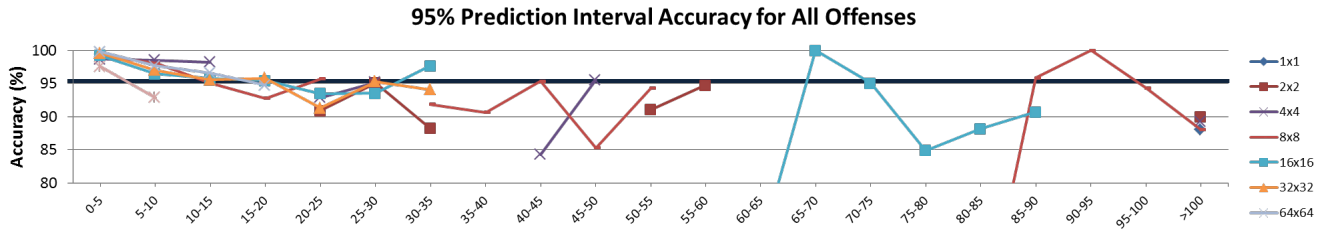


Fig. 7. 95% prediction interval accuracy vs. Avg(Y) for all CTC offenses for Tippecanoe County, IN. Here, geospace has been fragmented into rectangular grids of dimension $k \times k$ for various k values.

values), especially for regions with higher incidence rates. As can be seen from the results in Figures 6 and 7, the accuracy for higher $Avg(Y)$ values tend to be lower than the 95% prediction accuracy; thereby, indicating that the underlying data are slightly overdispersed. These results indicate that coarse scales can generate signals with too much variance, or combinations of multiple signals that overgeneralize the data. Furthermore, the signals generated at coarse scales can be affected by anomalies in underlying data (e.g., crime spikes during unusually high weathers, holidays). These can contribute to the non-Normality of the residuals, and produce an overdispersion of underlying data as compared to the assumptions of our model. It should be noted that although a slight data overdispersion is noticeable at coarse scales, they are deemed to be small enough to currently not warrant any correction. Finally, we note that further research is needed in order to determine the effects of these data overgeneralization issues at coarse scales and to devise strategies to mitigate for their effects.

7.4 Summary

Our model evaluation and validation strategy involved testing for the empirical rule of a Normal distribution where we tested whether the observed data conformed with the 95% prediction interval from our STL forecasting method at various geospatial scales. In order to cope with data sparseness issues, we performed our analysis at a weekly aggregation of data. Our results demonstrate the validity of our approach as long as the underlying assumptions of the underlying models are satisfied by the data. The results obtained using our DCKDE method are also promising. Our results also highlight the importance of performing analysis at appropriate scales, and demonstrate that the model predictions get severely biased when the underlying assumptions are violated by the data. We also explored the effects of data sparseness issues on our model predictions at fine geospatial scales. Our evaluation results show that the model predictions generated using input signals of 10 or more counts per time step on average tend to conform with the 95% prediction confidence intervals. We also highlight the effects of analysis performed at coarse scales, and show the data overgeneralization issues that occur at such scales. Although the results indicate a slight data overdispersion at coarse scales, the results show that the prediction accuracies from the model estimates still tend to converge at around the 95% confidence bounds. This further shows the effectiveness of our forecasting methodology in the geospatial domain. We also note that although our work enables hot spot policing and resource allocation strategy development, further evaluation is required to ascertain the efficacy of our predictive analytics framework when deployed in field. We leave this as future work.

8 Domain Expert Feedback

Our system was assessed by a police captain who oversees the operations and resource allocation of several precincts in a mid-sized police agency (of about 130 sworn officers) in the United States. In this section, we summarize the initial feedback received after conducting several informal interviews with him. The captain emphasized the need for a system that applies a data-driven approach to assist law enforcement decision makers in developing resource allocation strategies. He was impressed by the ability of the system to interactively generate various geospatial and temporal visualizations of historical datasets and forecast maps in real-time. Additionally, he also appreciated having the ability to dynamically apply any desired geospatial, temporal, and/or categorical filters on the data.

The captain stressed the need to carefully combine and aggregate different data categories for which reliable forecast maps could be generated. For example, he noted that a signal generated by combining two crime categories with different attributes (e.g., crimes against property and person) might introduce variability in the forecasting process and produce unreliable results. He further suggested that crimes of opportunity must be filtered out as these exhibit no discernable patterns. He asserted that different regions within the same city can exhibit different crime patterns due to the different underlying region dynamics. He expressed the importance for domain experts to create data category and spatiotemporal templates so viable prediction estimates can be computed using our methodology. Finally, the captain remarked that the predicted hotspot locations using aggregated CTC data occur at the known problem areas in the city.

9 Conclusions and Future Work

In this work, we have presented our visual analytics framework that provides a proactive decision making environment to decision makers and assists them in making informed future decisions using historical datasets. Our approach provides users with a suite of natural scale templates that support analysis at multiple spatiotemporal granularity levels. Our methods are built at the confluence of automated algorithms and interactive visual design spaces that support user guided analytical processes. We enable users to conduct their analyses over appropriate spatiotemporal granularity levels where the scale and frame of reference of the data analysis process and forecasting matches with that of the user's decision making frame of reference. It should be noted that while adjusting for the size of the geospatial and temporal scales is necessary, it is also important to adjust for the scale of the size of the dataset. A forecasting or analysis method that works well for one region with certain demographics and population densities may not have the same efficacy when applied to a different region. As such, our work explores the potential of visual analytics in providing a bridge so that different statistical and machine learning processes occur on the same scale and frame of reference as that of the decision making process.

Our future work includes developing new kernel density estimation techniques designed specifically for improving prediction forecasts. We further plan on improving our designed dynamic covariance kernel density estimation technique (DCKDE) to factor in for temporal distances to further enhance our STL based prediction algorithm. We also plan to incorporate data-driven methods that guide users in selecting between different choices provided by the system based on the underlying features of the data. We also plan on factoring in the influences and correlations among different variables to further refine our natural scale template generation methodology. Finally, we plan on conducting a formal user evaluation in order to understand the efficacy of our system in aiding domain experts to understand the properties of underlying data and their effects on the workings of the different underlying statistical processes.

Acknowledgments

This work was funded by the U.S. Department of Homeland Security VACCINE Center's under Award Number 2009-ST-061-CI0003.

REFERENCES

[1] G. Box and G. Jenkins. *Time series analysis: Forecasting and control*. Holden-Day, San Francisco, 1970.

[2] A. A. Braga. The effects of hot spots policing on crime. *Annals of the American Academy of Political and Social Science*, 578:pp. 104–125, 2001.

[3] A. A. Braga and B. J. Bond. Policing crime and disorder hot spots: A randomized controlled trial*. *Criminology*, 46(3):577–607, 2008.

[4] A. A. Braga, D. M. Hureau, and A. V. Papachristos. An ex post facto evaluation framework for place-based police interventions. *Police Quarterly*, 2012.

[5] C. A. Brewer. *Designing Better Maps: A Guide for GIS users*. ESRI Press, 2005.

[6] D. Brown and R. Oxford. Data mining time series with applications to crime analysis. In *IEEE International Conference on Systems, Man, and Cybernetics*, volume 3, pages 1453–1458 vol.3, 2001.

[7] J. S. d. Bruin, T. K. Cocx, W. A. Kosters, J. F. J. Laros, and J. N. Kok. Data mining approaches to criminal career analysis. In *Proceedings of the Sixth International Conference on Data Mining*, ICDM '06, pages 171–177, Washington, DC, USA, 2006. IEEE Computer Society.

[8] D. V. Canter. The environmental range of serial rapists. In D. V. Canter, editor, *Psychology in Action*, Dartmouth Benchmark Series, pages 217–230. Dartmouth Publishing Company, Hantshire, UK, January 1996.

[9] R. B. Cleveland, W. S. Cleveland, J. E. McRae, and I. Terpenning. Stl: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, 6:3–73, 1990.

[10] G. Cowan. *Statistical data analysis*. Oxford University Press, 1998.

[11] P. J. Diggle and P. J. Diggle. Statistical analysis of spatial point patterns. *London: Edward Arnold*, 1983.

[12] P. J. Diggle, J. Tawn, and R. Moyeed. Model-based geostatistics. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 47(3):299–350, 1998.

[13] G. Farrell and K. Pease. *Repeat victimization*, volume 12. Criminal Justice Press, 2001.

[14] M. Felson and E. Poulsen. Simple indicators of crime by time of day. *International Journal of Forecasting*, 19(4):595–601, 00 2003.

[15] J. S. Goldkamp and E. R. Vilcica. Targeted enforcement and adverse system side effects: The generation of fugitives in philadelphia. *Criminology*, 46(2):371–409, 2008.

[16] R. Hafen, D. Anderson, W. Cleveland, R. Maciejewski, D. Ebert, A. Abusalah, M. Yakout, M. Ouzzani, and S. Grannis. Syndromic surveillance: Stl for modeling, visualizing, and monitoring disease counts. *BMC Medical Informatics and Decision Making*, 9(1):21, 2009.

[17] K. Harries. *Crime and the environment*. American Lecture Series; No. 1033. Charles C. Thomas Publisher, Limited, 1980.

[18] S. D. Johnson, W. Bernasco, K. J. Bowers, H. Elffers, J. Ratcliffe, G. Rengert, and M. Townsley. Space–time patterns of risk: a cross national assessment of residential burglary victimization. *Journal of Quantitative Criminology*, 23(3):201–219, 2007.

[19] M. H. Kutner, C. J. Nachtsheim, J. Neter, and W. Li. *Applied linear statistical models*, volume 5. McGraw-Hill Irwin Chicago, 2004.

[20] R. Maciejewski, R. Hafen, S. Rudolph, S. Larew, M. Mitchell, W. Cleveland, and D. Ebert. Forecasting hotspots: A predictive analytics approach. *IEEE Transactions on Visualization and Computer Graphics*, 17(4):440 –453, April 2011.

[21] A. Malik, R. Maciejewski, T. F. Collins, and D. S. Ebert. Visual analytics law enforcement toolkit. In *IEEE International Conference on Technologies for Homeland Security*, pages 222–228, 2010.

[22] A. Malik, R. Maciejewski, N. Elmqvist, Y. Jang, D. Ebert, and W. Huang. A correlative analysis process in a visual analytics environment. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 33–42, 2012.

[23] M. Monroe, R. Lan, H. Lee, C. Plaisant, and B. Shneiderman. Temporal event sequence simplification. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2227–2236, 2013.

[24] J. D. Morenoff, R. J. Sampson, and S. W. Raudenbush. Neighborhood inequality, collective efficacy, and the spatial dynamics of urban violence*. *Criminology*, 39(3):517–558, 2001.

[25] T. Muhlbacher and H. Piringer. A partition-based framework for building and validating regression models. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):1962–1971, Dec 2013.

[26] D. A. Norman. *Things that make us smart: Defending human attributes in the age of the machine*. Basic Books, 1993.

[27] R. Oppenheim. Forecasting via the box-jenkins method. *Journal of the Academy of Marketing Science*, 6(3):206–221, 1978.

[28] A. Rind, T. Lammarsch, W. Aigner, B. Alsallakh, and S. Miksch. Timebench: A data model and software library for visual analytics of time-oriented data. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2247–2256, 2013.

[29] G. Robertson, D. Ebert, S. Eick, D. Keim, and K. Joy. Scale and complexity in visual analytics. *Information Visualization*, 8(4):247–253, 2009.

[30] L. W. Sherman. Police crackdowns: Initial and residual deterrence. *Crime and Justice*, 12:pp. 1–48, 1990.

[31] L. W. Sherman, P. R. Gartin, and M. E. Buerger. Hot spots of predatory crime: Routine activities and the criminology of place. *Criminology*, 27(1):27–56, 1989.

[32] M. Short, M. Dorsogna, P. Brantingham, and G. Tita. Measuring and modeling repeat and near-repeat burglary effects. *Journal of Quantitative Criminology*, 25(3):325–339, 2009.

[33] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall/CRC, 1986.

[34] S. J. South and S. F. Messner. Crime and demography: Multiple linkages, reciprocal relations. *Annual Review of Sociology*, 26(1):83–106, 2000.

[35] J. Stasko, C. Görg, and Z. Liu. Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization*, 7(2):118–132, 4 2008.

[36] S. S. Stevens. On the theory of scales of measurement, 1946.

[37] J. J. Thomas and K. A. Cook, editors. *Illuminating the Path: The R&D Agenda for Visual Analytics*. IEEE Press, 2005.

[38] W. R. Tobler. A computer movie simulating urban growth in the detroit region. *Economic geography*, pages 234–240, 1970.

[39] S. Towers. Kernel probability density estimation methods. *Proceedings of the Advanced Statistical Techniques in Particle Physics*, pages 107–111, 2002.

[40] P. F. Velleman and L. Wilkinson. Nominal, ordinal, interval, and ratio typologies are misleading. *The American Statistician*, 47(1):65–72, 1993.

[41] D. Weisburd, J. Hinkle, C. Famega, and J. Ready. The possible backfire effects of hot spots policing: an experimental assessment of impacts on legitimacy, fear and collective efficacy. *Journal of Experimental Criminology*, 7(4):297–320, 2011.

[42] L. Wilkinson and G. Wills. *The Grammar of Graphics*. Statistics and Computing. Springer, 2005.

[43] P. C. Wong, L. R. Leung, N. Lu, M. Paget, J. C. Jr., W. Jiang, P. Mackey, Z. T. Taylor, Y. Xie, J. Xu, S. Unwin, and A. Sanfilippo. Predicting the impact of climate change on u.s. power grids and its wider implications on national security. In *AAAI Spring Symposium: Technosocial Predictive Analytics*, pages 148–153. AAAI, 2009.

[44] C.-H. Yu, M. W. Ward, M. Morabito, and W. Ding. Crime forecasting using data mining techniques. In *Proceedings of the IEEE 11th International Conference on Data Mining Workshops*, ICDMW '11, pages 779–786, Washington, DC, USA, 2011. IEEE Computer Society.

[45] J. Yue, A. Raja, D. Liu, X. Wang, and W. Ribarsky. A blackboard-based approach towards predictive analytics. In *AAAI Spring Symposium: Technosocial Predictive Analytics*, page 154. AAAI, 2009.

Real-time Identification and Monitoring of Abnormal Events Based on Microblog and Emergency Call Data Using SMART

Jiawei Zhang, Shehzad Afzal, Dallas Breunig, Jing Xia, Jieqiong Zhao, Isaac Sheeley, Joseph Christopher, David S. Ebert, *Fellow, IEEE*, Chen Guo, Shang Xu, Jun Yu, Qiaoying Wang, Chen Wang, Zhenyu Qian, Yingjie Chen

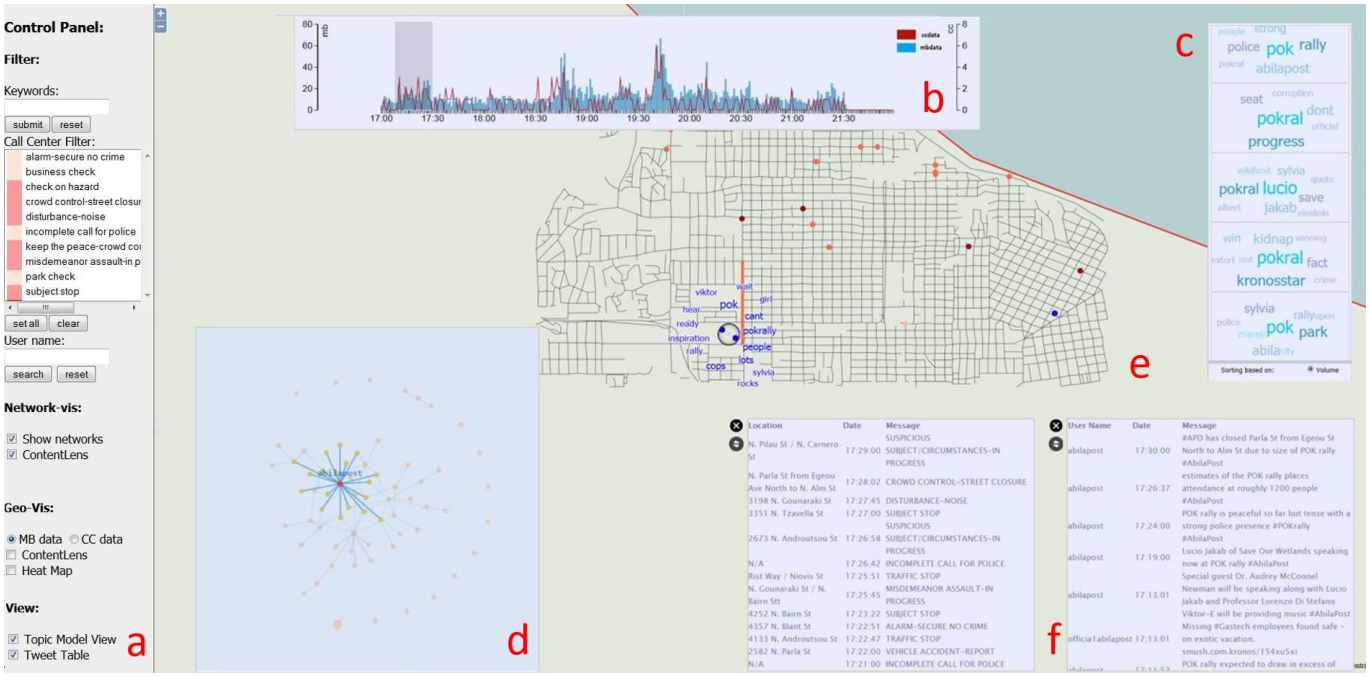


Fig. 1. Our system helps the analyst monitor POK Rally, which is held in Abila City Park. The system consists of a) control panel; b) time series view; c) LDA topic view; d) reply/retweet networks view; e) map view and f) microblog & emergency call table. Through interactive and visual exploration, several major events occurred in the selected time period are revealed, including: 1) Part of Parla St was closed due to concerns of public safety (map view); 2) several key roles appeared in the rally including Sylvia Marek and Lucio Jakab (topic view); 3) some news media, such as *abilapost*, actively posted updates of the rally (networks view).

Abstract— This article describes a real-time visual analytics process based on microblog and emergency call data to solve VAST 2014 Mini Challenge 3. We extended SMART system (Social Media Analytics and Reporting Toolkit), developed by the U.S. Department of Homeland Security's VACCINE Center. Our system consists of multiple linked views to allow the analyst monitor topic evolution, identify influential microblog users, observe geo-location patterns and examine correlations among different data sources. Extensions to our previous work include a time series view, a reply/retweet networks view, and integration of emergency call data.

Index Terms—Visual Analytics, Microblog, Spatiotemporal Analysis, Multiple Linked Views

1 Introduction

Given real-time microblog and emergency call data stream, it is crucial for investigators to combine multiple data sources to identify abnormal events and build connections among different events in a timely manner. In such scenarios, we summarize the following major tasks which serve instrumental roles in the investigation process:

- An integral framework involving analytics of multiple data attributes, including timestamp, geo-location, text (messages) and relationship structure (reply/retweet).
- Effective retrieval of relevant messages from large volume data stream.
- Support of correlation analysis between microblog and emergency call data.

We utilized and extended our previous work, SMART system. SMART is a web-based framework which captures real-time social media stream and allows analysts perform topic extraction, aggregate and visualize major keywords, and categorize messages related to different events. Based on SMART, we incorporated a time series view to visualize data volume trends and a networks view to visualize reply/retweet relationships among microblog users. In the following

- *Jiawei Zhang, Shehzad Afzal, Dallas Breunig, Jieqiong Zhao, Isaac Sheeley, Joseph Christopher, David S. Ebert, Guo Chen, Shang Xu, Jun Yu, Qiaoying Wang, Zhenyu Qian and Yingjie Chen are with Purdue University in West Lafayette, IN, USA. E-mail: {zhan1486, safzal, dbreunig, jieqiongzhao, isheeley, christ33, ebertd, guo171, xu537}@purdue.edu, yujun_2005@hotmail.com, {wang1925, qianz, victorchen}@purdue.edu.*
- *Jing Xia is with Zhejiang University in Hangzhou, China. E-mail: xiajing@zjucadcg.cn.*
- *Chen Wang is with Harbin Institute of Technology in Harbin, China. E-mail: wangchen@hit.edu.cn.*

IEEE Symposium on Visual Analytics Science and Technology 2014
November 9-14, Paris, France
978-1-4799-6227-9/14/$31.00 ©2014 IEEE

sections, we focus on the above mentioned tasks and demonstrate the capabilities of our system to effectively perform investigation analysis.

2 Visual Analytics Approach

2.1 Multiple Linked Views

In order to understand events of interest to the maximum extent and avoid bias, we utilize the following multiple linked views to visualize a subset of data specified by the analyst.

- **Time Series View** visualizes temporal trends of data volume over time. Microblog data is rendered using blue bar charts and emergency call data is rendered using red line charts.
- **Topic View** visualizes topics extracted using LDA [1] model. Each rectangle represents a topic described by a set of keywords.
- **Networks View** visualizes reply/retweet dynamic networks using force-directed layout [3]. Nodes represent users and edges represent reply/retweet relationships.
- **Map View** visualizes geo-tagged microblog data and emergency call data, the geo-location of which is parsed in the back-end side based on the given KML map file. Microblog records are rendered using blue dots. Emergency call records are parsed as dots or lines based on their semantic structure, and then we manually assign impact factors based on different types of emergency calls and render them using a sequential color scheme, as is shown in the control panel, in Fig.1.
- **Microblog & Emergency Call Table** visualizes detailed microblog and emergency call records including user handle/emergency location, timestamp and message content.

The analyst can easily specify data of interest through interactions including 1) selecting time range in the time series view; 2) clicking a topic in the topic view and 3) selecting one or more users in the networks view using a circular shape. The linked views then automatically refresh to highlight the corresponding data sample.

The analyst can also click a single record in the microblog & emergency call table. The time series view highlights the corresponding time bar. If the record is geo-tagged, the map view highlights the corresponding geo-location using a hot dot animation (Fig.2).

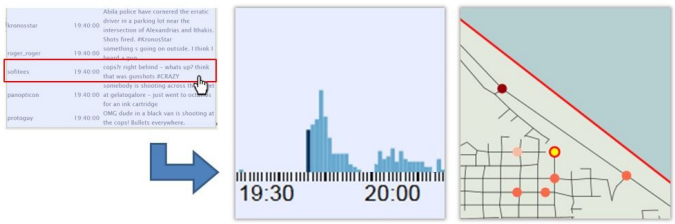


Fig. 2. When the analyst clicks a record in the message table, the time series view and the map view highlight the selected record respectively.

Through multiple linked views, the analyst is able to simultaneously investigate, compare and validate based on different dimensions and perspectives of data stream.

2.2 Locating Relevant Messages in Large Volume Data Stream

Massive data stream with a large portion of irrelevant messages hinders effective identification and monitoring of events of interest. Based on multiple linked views and interactive filtering in individual views, the analyst is allowed to iteratively filter data, refine results and drill down to locate more relevant information.

In the time series view, the analyst can perform filtering by selecting the time range containing volume burst, which can probably indicate a certain abnormal event occurs within that period of time.

In the topic view, the analyst is able to monitor ongoing topics and by clicking a topic of interest, the system triggers filtering based on keywords associated with the selected topic.

In the networks view, assuming users of high degree centrality serve influential roles in disseminating and communicating information in the social network structure, the analyst can perform filtering based on a group of key roles to track their conversation and interaction for more relevant and reliable updates of events.

In the map view, the analyst can also perform spatial filtering to discover location-based patterns.

Hence, effective filtering of individual components and combination of filters supported by the system allow the analyst quickly and accurately capture meaningful information from massive data.

2.3 Support of Correlation Analysis between Microblog and Emergency Call

Microblog and emergency call data have similar attributes, such as timestamp, geo-location and message content (a small portion of microblog messages and all emergency call records are geo-tagged.). We mainly focus on revealing the temporal and spatial correlations through our visual analytics approach.

In the time series view, as is shown in Fig.1, two different types of data have very similar pattern of evolution over time, indicating the occurrence of an abnormal event is usually followed by increasing emergency calls and active posts in microblog traffic. Hence, it is reasonable to regard data volume burst as an early signal for event identification.

In the map view, we render two types of data based on different color schemes (mentioned in Section 2.1). By using contentlens[1], heat map and performing spatial filtering, the analyst is able to restrict data to detailed level of geo-location, combine and examine different data sources to discover location-based correlation.

Incorporating analysis of temporal and spatial correlations, the analyst is able to reveal underlying connections among different events. For example, the shooting & standoff event happened immediately after the hit & run event in the timeline. In both cases the perpetrators were driving a black van and pursued down Egeou Avenue at the same time and in the same direction, which leads to the deduction that the same group of perpetrators engaged in the two events.

3 Conclusion

As an extension of SMART system, we present an integrated visual analytics framework to incorporate multiple data sources and data attributes for event identification and monitoring using microblog and emergency call stream. By using the proposed system, we successfully identified several major events happened in Abila city, monitored their timelines, and investigated their underlying connections. Some future work includes more efficient layout and analytics functionality in networks view, incorporating data of multiple social media services for more reliable information retrieval, etc.

Acknowledgments

This work was partially funded by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0003.

References

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. In *NIPS*, pages 601–608, 2001.

[2] H. Bosch, D. Thom, F. Heimerl, E. Puttmann, S. Koch, R. Kruger, M. Worner, and T. Ertl. Scatterblogs2: Real-time monitoring of microblog messages through user-guided filtering. *Visualization and Computer Graphics, IEEE Transactions on*, 19(12):2022–2031, 2013.

[3] T. M. J. Fruchterman and E. M. Reingold. Graph drawing by force-directed placement. *Softw. Pract. Exper.*, 21(11):1129–1164, Nov. 1991.

[1] Contentlens: A visualization technique applied to SMART, which shows the most prominent terms in geo-tagged messages within a certain area by moving a circular shape. [2]

Safety in View: A Public Safety Visual Analytics Tool Based on CCTV Camera Angles of View

Hanye Xu, James Tay
Purdue University

Abish Malik, Shehzad Afzal
Purdue University

David S. Ebert
Purdue Univeristy

***Abstract*—Campus security and police departments have implemented a multitude of safety precautions, including CCTV cameras. The efficiency and effectiveness of using CCTV camera resources for preventing crimes result in higher demand. We implemented a visual analytics tool to analyze the existing CCTV camera resources and suggest improved allocation schemas based on blind spots and crime data. Our tool provides the user with an interactive safe path calculation method for walking purpose on the basis of the maximum monitoring area. Additionally, avoiding buildings in the calculated path is an optional control factor. Our tool also provides functions for crime data analysis. The camera-alarming function highlights the cameras that a specific crime occurred in their visible range. The camera-ranking function highlights the camera that records the largest number of crime incidents. Based on the historical crime data, we suggest locations for future camera installation. We present two case studies to illustrate the usage and features of our tool on the campus of Purdue University.**

I. Introduction

Campus safety is a serious concern for both students and parents when choosing an educational institution. Educational institutions invest substantial resources and funding to maintain campus safety. Therefore, more high-quality surveillance cameras are being installed on campuses to monitor safety of patrons. For example, according to the record of Purdue University Police Department, more than 100 Closed Circuit Television (CCTV) cameras were installed in 2014 comparing to 54 CCTV cameras in 2010 [1]. With the increasing number of CCTV cameras and a larger camera viewing angle datasets, there is a shortage of analysis tools for both police and pedestrians to make good use of such datasets. This paper introduces a visual analytical tool that uses camera location and coverage data to enable law enforcement decision makers to assess and improve pedestrian safety.

The tool presented in this paper has been designed for two user groups: pedestrians and campus security departments. Pedestrians can utilize our tool to determine the shortest walking path between a user-provided origin and destination such that CCTV cameras maximally cover this path. Our tool provides users dragging interactions to adjust path solutions on a map. Our approach attempts to minimize the time delay caused as a result of diverging from the shortest path to improve camera coverage. For campus security departments, we provide crime data analysis functions based on camera coverage. Our tool allows users to quickly find cameras that capture specific crimes. Our system also uses historical crime data to provide importance-based rankings of all cameras. This feature enables police agencies to prioritize the cameras based on the number of incidents that were captured by each camera historically; thereby, providing them with suggestions on critical cameras to focus on. We also include a feature that provides suggestions of locations for installing new cameras based on historical crime data. The main contributions of this work are the following:

1) An interactive visual analytics tool that supports safe path calculations.
2) The integration of crime data and camera coverage data to facilitate decision-making and mitigation of risk.

The remainder of this paper has been structured in the following sections. We introduce related work in the Section2. Detailed analysis methods and solutions are described in Section 3. Section 4 provides two case studies. Finally, we conclude our work and discuss further goals for our system in Section 5.

II. Related work

In recent years, a large number of geospatial applications have been developed for the desktop, web, and mobile devices [2]. In this section, we summarize past work in path optimization, public safety data analysis, and camera analytics.

A. Paths and routes visualization on map

Map visualizations have been used for hundreds of years to show objects in a space. With the development of digital technology, online maps are frequently used in our daily life. In order to make good use of maps, researchers have focused on path and route design and made good contributions to modern transportation systems [3]. Also, algorithms have been developed to optimize paths on a map [4] in order to generate optimal routes. Routing applications have also been developed in recent years [5] that improve safety for patrons by focusing on locating spatial obstacles in order to reduce accidents. Some recent studies also provided route representations based on user feedback [5]. These visualization tools also provide users with several interaction methods that allow them to customize their routes. Accordingly, our system also integrates rerouting for generating a safe walking path.

B. Public safety data analysis

Public safety is always a big challenge for modern societies. Making efficient use of data remains a difficult issue. According to recent work, many researchers have developed

978-1-4799-1737-2/15/$31.00 ©2015 IEEE

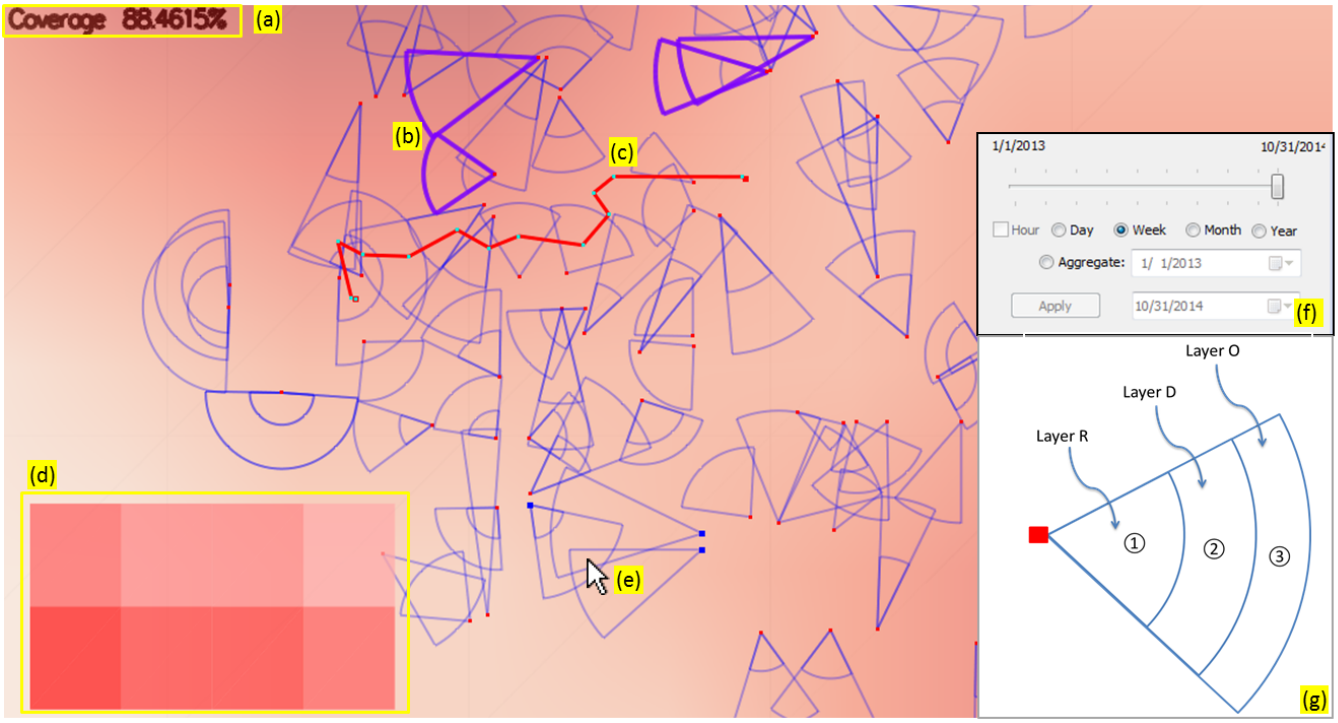


Fig. 1. Screenshot of our Safety in View tool that safe paths for pedestrians, and suggests new CCTV camera installation locations for law enforcement agencies.

analysis tools that facilitate public safety data analysis. For example, a visualization tool makes use of the locations of 911 calls in Seattle to show the geo-location of callers [6]. Also, previous historical crime studies [10] and visualizations show significant results for crime prevention analysis and decision-making process. The system of Razip et al. [8] provides us with an example of how public crime data can be effectively used to make informed future decisions based on historical crime datasets. A summary of crime mapping and spatial statistics can be found in [7, 9]. This previous research has shown that data from the public can be better analyzed with visualization tools, such as ours, for improved crime prevention. Our tool is also able to incorporate this crime data and generate safe pathways as well as facilitate analysis and decision-making processes.

C. Camera video visualization

Security and surveillance cameras are widely used today. In recent years, many data analysis tools have been developed to analyze video data. Several video visualization tools have been developed to facilitate video information perception, such as Video Visualization by Chen et al. [11]. Another visualization tool called LiveLayer has been developed to improve the real-time traffic analysis process [12]. Our system makes use of camera viewing angles datasets to facilitate crime prevention. In future, we will incorporate video visualization and real-time analysis into our current system to elevate efficiency and quality of the crime analysis process.

III. Our Public Safety Analysis Tool

The public safety toolkit presented in this paper has been developed on our existing framework that provides the user with tools to interactively visualize and analyze historical criminal, traffic, and civil (CTC) incidents in different geo-spatial and temporal displays [14]. A snapshot of our system is shown in Figure 1.[1] The main view of our system is the map view that overlays the camera coverage along with the historical distribution of incidents on the map. The calculated safe path (i.e., the path that maximizes the camera coverage) is shown on the map in red in Figure 1(c). The locations of all cameras are shown as red dots on the map. When the mouse cursor is placed over the camera viewing angles, the corresponding camera locations are shown in blue small squares in Figure 1(e). The camera that captures the most number of crimes for the user specified time period is highlighted on the map in Figure 1(b). Figure 1(f) shows the interactive time selection window that allows users to temporally slide through their CTC data at various data aggregation levels (e.g., by day, week, month, year).

A. Safe path calculation

Our analysis tool provides users with two styles of interactive safe walking path solutions to improve pedestrian safety. Figure 2 (a) shows the original path obtained from the Bing Map API from starting point (S) to destination

[1]In the figures, the actual campus map layer has been removed to protect the locations of the actual cameras that are not publicly known.

978-1-4799-1737-2/15/$31.00 ©2015 IEEE

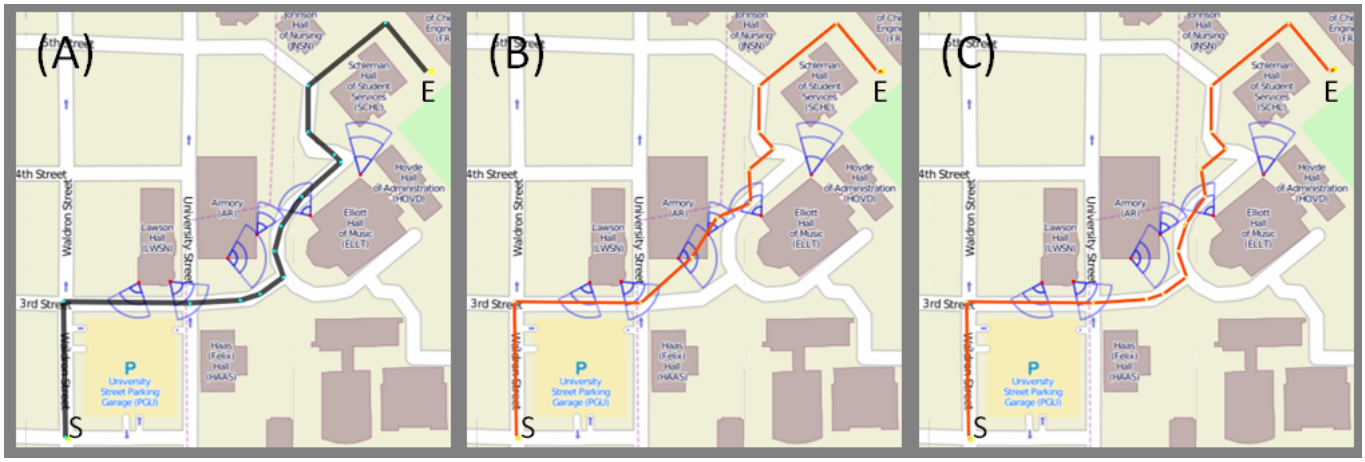


Fig. 2. A comparsion between a map generated path between two locations (A), an optimized safe path doing through buildings (B), and a safe path following roadways (C). For illustration purposes, notional camera locations were used in the calculation and for display.

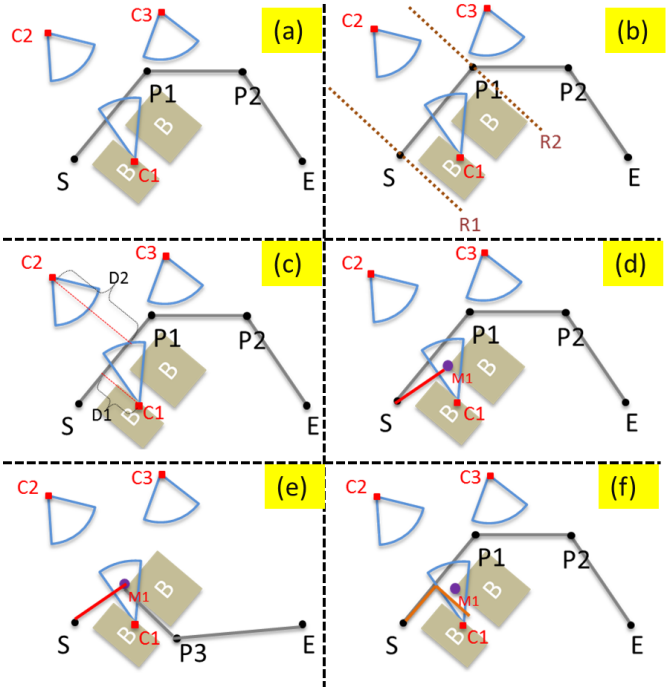


Fig. 3. An illustration of the steps of safe path calcuation process.

point (E) on the map. The first style (Figure 2 (b)) is a short path disregarding buildings to maximize possible camera monitoring path segments. The second style (Figure 2 (c)) is the shortest path that ensures the walking path always remains on roadways. The selection between two styles of safe path is provided on the menu. Our analysis tool also allows the user to select a time delay threshold value that sets the maximum deviation allowed from the shortest path. For example, if the user sets the maximum deviation to be 3 minutes and the original path takes 10 minutes, then the new calculated path will take less than a total of 13 minutes to arrive at the desired destination. According to the specifications of CCTV cameras on campus, in Figure 1 (g), each camera has up to three viewing ranges. A human face is recognizable in the smallest range R. Gestures are recognizable in the second viewing range (layer D). In range O, only movements are identifiable. Additionally, the system allows the user to drag points along the safe path in order to modify the path to their desired route.

Safe path calculation:

In order to calculate a safe path between the user-provided source and destination points, the system first utilizes the Bing Map API to obtain the original walking path. The obtained results consist of waypoints in between the source and the destination. Figure 3 (a) shows one example path obtained from the Bing Map API, where S and E stand for the start and end points of the path, respectively. Every two adjacent points are treated as a sub path that is used to search for the nearest camera location. Next, boundaries are defined for all subsequent waypoints in order to find the nearest camera locations for the sub-path. This is shown in Figure 3 (b), where the region between lines R1 and R2 forms the boundary between points S and P1. This boundary filters out a large proportion of cameras that are not close to this sub path. As can be seen from Figure 3 (b), cameras C1 and C2 are within this boundary. Consequently, cameras C1 and C2 are selected to be candidate cameras. Next, the system computes the perpendicular distance from the cameras to the path within the boundary region (D1 and D2 in Figure 3(c)). The camera closest to the sub-path is then selected by the system based on the perpendicular distance, provided the distance is within the selected time delay range (camera C1 is picked in Figure 3(c)). In Figure 3(d) and 3(f), we show the comparison between two different styles of safe paths calculation, the casual style of safe path and the theoretical style of safe path.

The safe path disregarding buildings chooses the center the camera viewing angles (Figure 3(d)) M1 as a next turning point. Then, the generated safe path shown in red goes directly to M1. However, the safe path avoids the buildings

978-1-4799-1737-2/15/$31.00 ©2015 IEEE

to obtain a new walking path from S to M1, because Bing Map API always provides paths on a roadway (Figure 3(f)). By directly using the acquired path, the generated safe path is shown in orange in Figure 3(f). Next, if we use the first style (Figure 3(d)) safe path to continue our calculation, M1 is treated as new turning point and used to acquire a new ordinary path from M1 to E. In Figure 3(e), the new original walking path is in grey. Its first sub path from M1 to P3 does not have any close camera in range. So P3 is chosen to be the next starting point for acquiring a new walking path. By repeating these steps, our system is able to generate a safe path for walking within camera view ranges. The recursive calculation terminates when the ordinary walking path has no more turning points, or when the destination point is much shorter than the acquired ordinary walking path.

Path optimization:

After we obtain the safe path with camera coverage, we also optimize the path based on user interaction by allowing the user to select start and destination points anywhere on the map including on buildings. The Bing Map API does not recognize points inside buildings, and defaults these points to the nearest roadway. The first path optimization we implemented connects the user-clicked starting and ending points with the calculated safe path. The second implementation we added is the avoidance of looping all the way around a structure when one can take a more direct path.

Coverage percentage calculation:

After optimizing the calculated safe path, our tool shows the percentage of camera coverage along the generated path (as shown in Figure 1(A). In Figure 4 (A), the percentage is *l2/ (l1+l2+l3)*, since only the region labeled as *l2* is within the coverage of a camera.

Detection of a point inside a sector:

We use a cross-product based method to detect points in a sector (A cross B). First, if point D is inside a sector, the cross product result of vector AB and AD is negative and the cross product result of vector AC and AD is positive (Figure 4(B)). If the point meets the requirement, then the distance from D to A is calculated. If the length AD is shorter than this sectors radius, then the point D is considered as a point inside of this sector.

B. Crime data analysis

Our analytics tool also provides three crime data analysis features. The current work extends our crime hotspot visual analytics work [14]. Based on camera coverage data and historical crime data, our tool is able to provide camera locations relative to a specific crime, suggests locations for future camera installation locations, and ranks all cameras based on importance in capturing historical crime activity.

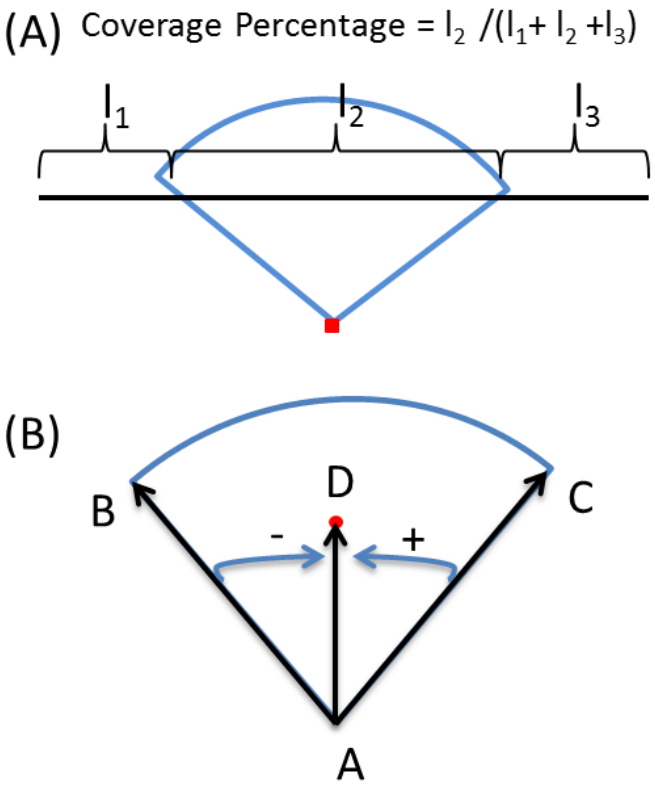


Fig. 4. Illustrations for coverage percentage calculation and detection of a point inside a sector.

Importance based camera ranking:

Since a large number of cameras need to be monitored at the same time, importance based ranking is beneficial by providing a quick guide on which specific cameras to monitor at a specific time. Figure 1(b) shows violet outlines of sectors that represent those camera view ranges that contain the largest number of criminal incidents in the past year. As in our previous work, crime incidents chosen by the user are updated as the user scrolls through time (Figure 1(f)). With the filtered crime incident locations, we detect the total number of incidents that happened in every cameras view range and use this for determining the cameras importance.

Suggestions for future camera location:

In order to make efficient use the historical crime database for monitoring, our system provides the user with suggestions for future camera setup locations. By categorizing historical criminal incidents by their locations, our tool determines the locations of high frequency occurrences. If no camera monitors such a location, then we provide suggestions of locations to install a new CCTV camera. This feature is based on the past years crime database. By using this result from VALET [13], we categorize crime by locations according to grids on the map. As can be seen in Figure 1(d), two grids adjoin to each other on the map. Each grid consists of four small grids. Colors

978-1-4799-1737-2/15/$31.00 ©2015 IEEE

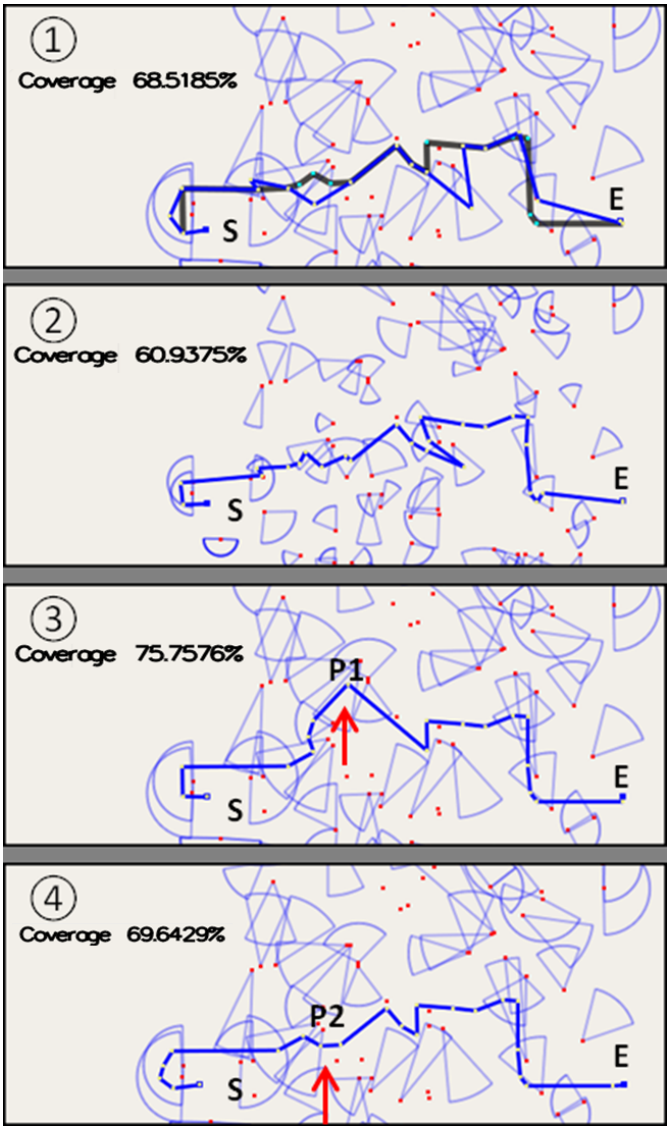


Fig. 5. Images highlighting the steps of case study 1.

are assigned to grids based on the number of crime occurred in the grid area. So the darkest small grid provides suggested camera installation location.

IV. Case Study

In this section, we demonstrate our system by utilizing Purdue Universitys database of CCTV cameras. Each camera has up to three different visibility region layers as discussed in Section 3. The CTC incident datasets are obtained from reported incidents in Tippecanoe County, IN, U.S.A. This database contains spatiotemporal crime information. We provide two examples that demonstrate the usability and specific features of our visual analytics system. We also discuss methods for preventing crimes and improving pedestrian safety by using our system.

A. Case study 1: Safe path analysis

In this example, we provide a hypothetical scenario where the user is interested in finding out the shortest safe path from the starting location S to the ending location E. After the camera shape file is loaded in the system, the user is able to draw a path on the map, selecting the starting location S and the ending location E. From the menu, the user is able to show both the original path and the calculated short path on the map. In Figure 5(1), the black path shows the original walking path, and the blue path is the generated shortest safe path.

When the user chooses to use the smaller camera viewing range of all cameras on campus (e.g., Figure 5(2)), the coverage percentage decreases, and the generated path is changed to be closer to the camera locations. This feature is especially useful for situations where the weather is bad, and visibility of camera coverage is low. Furthermore, if there is blockage on a path, such as closure due to a car accident on the generated path, the user can interactively modify the safe path by dragging the safe path to another location on the map. This operation is shown in Figures 5(3 and 4), where the user has modified the waypoint P1 to P2. The percentage of camera-monitored path segments also changes as a result of this operation, and is shown to the end-user.

B. Case study 2: Crime Data Analysis

In our second example, we demonstrate how our system uses historical crime incident data to provide safety recommendations to the user. We assume that the user selects all crime incidents in our database to use in the analysis. In Figure 1 (b), a sector is highlighted in violet to show the camera that records the most number of crimes on campus for a user selected time range. The user can use this information to prioritize monitoring the highlighted camera in order to prevent possible crime activities in that area.

When the user hovers over a location, camera locations that fall under the cursor are highlighted to notify the user which cameras have the video records of that location (e.g., Figure 1(e). This is especially useful since most surveillance systems do not provide the ability to identify the cameras that cover a user selected region. In this scenario, the surveillance videos for the highlighted cameras are checked first for searching and obtaining evidence. We envision this feature also be used for real time tracking purposes. For instance, if suspicious behavior is detected by a surveillance camera, the user can use the system to locate other cameras on the map that provide overlapping coverage for further investigation.

Finally, we note that as surveillance systems are often purchased incrementally, we also incorporate the feature that helps law enforcement agencies to determine the location for installing a new surveillance CCTV camera. The user can use our system to visualize all incidents for the selected time period. The system then uses the selected data subset to highlight the areas on the map where the highest number of incidents occurred that were not covered by current cameras. This is shown in Figure 1(d), where the two dark colored grids

978-1-4799-1737-2/15/$31.00 ©2015 IEEE

on the map are the suggested camera installation locations for new cameras. Law enforcement agencies can therefore utilize our system to determine high incident areas with no camera coverage for installing new cameras.

V. Conclusion and future work

In this work, we demonstrate a visual analytics system for a safe path, shortest path analysis and crime analysis based on specific safety concerns. These interactive visual representations allow the user to obtain solutions for their safety. Our system provides users with maximized CCTV camera coverage for the shortest walking path with calculated camera coverage percentages. It also allows the user to customize the suggested solution to meet his or her specific needs. Moreover, our system provides the user with a camera finder feature to locate a specific camera, to identify crime related camera rankings that improve monitoring efficiency, and provide solutions for future installment of CCTV cameras.

In real life situations, it is important for security departments to maintain safety in campus areas. In addition, preventing crime incidents and decreasing the possibilities of crime can be more effective than simply monitoring campus areas. Our system makes effective use of CCTV camera data to assist security departments in preventing possible crimes. We also present two case studies to better illustrate the features and usage of our visual analytics system that respond to the needs described to us by our local law enforcement and public safety partners. In those examples, we detailed usages and realistic applications for our system on Purdue Universitys campus.

In the future, we aim to improve this system as a web based application that will be more accessible for the general public across campus. Additionally, this system is able to extend to a larger area of analysis moving forward. For example, it could extend to city CCTV cameras and crime analysis if we are given permission to access those databases.

Acknowledgment

The authors would like to thank the Purdue University Police Department for their support with this project. This work was funded by the U.S. Department of Homeland Security VACCINE Centers under Award Number 2009-ST-061-CI0003.

References

[1] Weddle, E, “Purdue installing exterior security cameras.”, Retrieved June, 2014. Available: http://www.jconline.com/article/20100309/NEWS0501/100308024.

[2] Eick, S.G.; Eick, M.A.; Fugitt, J.; Horst, B.; Khailo, M.; Lankenau, R.A., “Thin Client Visualization.” IEEE Symposium on Visual Analytics Science and Technology, 2007., vol., no., pp.51,58, Oct. 30 2007-Nov. 1 2007.

[3] Kray, C.; Elting, C.; Laakso, K.; Coors, V., “Presenting route instructions on mobile devices. In Proceedings of the 8th International Conference on Intelligent User Interfaces (IUI), ACM (2003), 117124.

[4] Goczyla, K.Cielatkowski, J.“Optimal Routing in a Transportation Network.”, European Journal of Operational Research, 199587, 214-222.

[5] Reddy,S.; Shilton, K.; Denisov, G.;Cenizal, C.; Estrin, D.; Srivastava,M. B., “Biketastic: Sensing and mapping for better biking., in Proc.SIGCHI Conf. Human Factors Compute. Syst., 2010, pp. 18171820.

[6] Duke, J., “Visualizing Seattle’s 911 calls.”, Retrieved June, 2014. Available: http://jmduke.com/posts/visualizing-seattles-911-calls/

[7] Eck, J.E.; Chainey, S.; Cameron, J.G.; Leitner, M.; Wilson, R.E., “Mapping crime: Understanding hotspots.”, 2005,pp.1-71.

[8] Razip, A.M.M.; Malik, A.; Afzal, S.; Potrawski, M.; Maciejewski, R.; Yun Jang; Elmqvist, N.; Ebert, D.S., “A Mobile Visual Analytics Approach for Law Enforcement Situation Awareness.”, 2014 IEEE Pacific Visualization Symposium (PacificVis), vol., no., pp.169,176, 4-7 March 2014.

[9] Luc, A.; Cohen, J.; Cook, D.; Gorr, W.; Tita, G., “Spatial analyses of crime.”, Criminal justice 4, no. 2 (2000): 213-262.

[10] Hsinchun, C.; Chung, W.; Xu, J.J.; Wang, G.; Qin, Y.; Chau, M., “Crime data mining: a general framework and some examples.”, Computer, vol.37, no.4, pp.50,56, April 2004.

[11] Daniel, G.; Chen, M., “Video visualization.”, Visualization, 2003. VIS 2003. IEEE, vol., no., pp.409, 416, 24-24 Oct. 200.

[12] Walton, S.; Chen, M.; Ebert, D., “LiveLayer: Real-time Traffic Video Visualization on Geographical Maps”, Available:https://www.purdue.edu/discoverypark/assets/pdfs/publications/pdf/Real-time%20Traffic%20Video%20Visualisation%20on%20Geographical%20Maps.pdf

[13] Malik, A.; Maciejewski, R.; Elmqvist, N.; Jang, Y.; Ebert, D.S.; Huang, W., “A correlative analysis process in a visual analytics environment.”, Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on , vol., no., pp.33,42, 14-19 Oct. 2012

[14] Malik, A.; Maciejewski, R.; Collins T. F.; Ebert, D.S., “Visual analytics law enforcement toolkit.”, In Proceedings of IEEE Conference on Technologies for Homeland Security, pages 222228, 2010.

978-1-4799-1737-2/15/$31.00 ©2015 IEEE

VASA: Interactive Computational Steering of Large Asynchronous Simulation Pipelines for Societal Infrastructure

Sungahn Ko, Jieqiong Zhao, Jing Xia, *Student Member, IEEE*, Shehzad Afzal, Xiaoyu Wang, *Member, IEEE*, Greg Abram, Niklas Elmqvist, *Senior Member, IEEE*, Len Kne, David Van Riper, Kelly Gaither, Shaun Kennedy, William Tolone, William Ribarsky, David S. Ebert, *Fellow, IEEE*

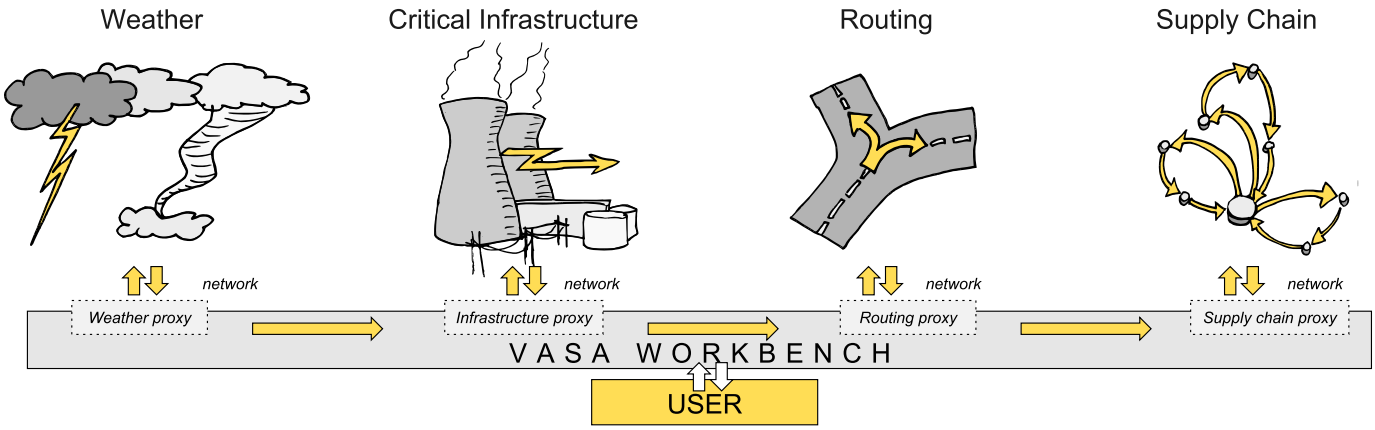


Fig. 1. Conceptual overview of the VASA system, including four simulation components for weather, critical infrastructure, road network routing, and supply chains, as well as the VASA Workbench binding them together.

Abstract—We present VASA, a visual analytics platform consisting of a desktop application, a component model, and a suite of distributed simulation components for modeling the impact of societal threats such as weather, food contamination, and traffic on critical infrastructure such as supply chains, road networks, and power grids. Each component encapsulates a high-fidelity simulation model that together form an asynchronous simulation pipeline: a system of systems of individual simulations with a common data and parameter exchange format. At the heart of VASA is the Workbench, a visual analytics application providing three distinct features: (1) low-fidelity approximations of the distributed simulation components using local simulation proxies to enable analysts to interactively configure a simulation run; (2) computational steering mechanisms to manage the execution of individual simulation components; and (3) spatiotemporal and interactive methods to explore the combined results of a simulation run. We showcase the utility of the platform using examples involving supply chains during a hurricane as well as food contamination in a fast food restaurant chain.

Index Terms—Computational steering, visual analytics, critical infrastructure, homeland security.

1 Introduction

Highways, interstates, and county roads; water mains, power grids, and telecom networks; offices, restaurants, and grocery stores; sewage, landfills, and garbage disposal. All of these are critical components of our societal infrastructure that help run our world. However, the complex and potentially fragile interrelationships connecting these components also mean that this critical infrastructure is vulnerable to both natural and man-made threats: twisters, hurricanes, and flash floods; traffic, road blocks, and pile-up collisions; disease, food poisoning, and major pandemics; crime, riots, and terrorist attacks. How can a modern society protect its critical infrastructure against such a diverse range of threats? How can we design for resilience and preparedness when perturbation in one seemingly minor aspect of our infrastructure may have vast and far-reaching impacts across society as a whole?

Simulation, where a real-world process is modeled and studied over time, has long been a standard tool for analysts and policymakers to answer these very questions (e.g., applications for modeling the real world [10]). Using complex simulations of critical infrastructure components, expert users have been able to create "what-if" scenarios, calculate the impact of a threat depending on its severity, and—last but not least—study optimal mitigation measures to address them. In fact, analysts have gone so far as to name "simulation as the new innovation" [35]: instead of endeavoring to produce the perfect solution once and for all, this new school of thought is to create a whole range of possible solutions and determine the optimal one using modeling and simulation. For example, during the Obama reelection campaign, it was reported that Organizing for Action data analysts ran a total of 62,000 simulations to determine voter behavior based on data from social media, political advertisements, and polling [43]. Basically, the philosophy with big data analytics driven by simulation is not to get the answer perfectly right, but to be less wrong over time [34]. Put differently, while it would be inappropriate to state—as others have done [2]—that big data will never somehow overtake theory, it is clear

- *Sungahn Ko, Jieqiong Zhao, Shehzad Afzal, Niklas Elmqvist, and David S. Ebert are with Purdue University in West Lafayette, IN, USA. E-mail: {ko, zhao413, safzal, elm, ebertd}@purdue.edu.*
- *Jing Xia is with Zhejiang University in Hangzhou, China. E-mail: xiajing@zjucadcg.cn.*
- *Xiaoyu Wang, William Tolone, and William Ribarsky are with University of North Carolina at Charlotte in Charlotte, NC, USA. E-mail: {xiaoyu.wang, ribarsky}@uncc.edu.*
- *David Van Riper, Len Kne and Shaun Kennedy are with University of Minnesota in Minneapolis, MN, USA. E-mail: {vanriper, lenkne, kenne108}@umn.edu.*
- *Greg Abram and Kelly Gaither are with University of Texas at Austin in Austin, TX, USA. E-mail: {gda, kelly}@tacc.utexas.edu*

Submitted to IEEE VAST 2014. Do not redistribute.

that large-scale simulation is a new and powerful tool in our arsenal for making sense of the world we live in.

Applying simulation to the scope of entire critical infrastructures—such as transportation, supply chains, and power grids—as well as the factors impacting them—such as weather, traffic, and man-made threats—requires constructing large *asynchronous simulation pipelines*, where the output of one or more simulation models becomes the input for one or more other simulations arranged in a sequence with feedback. Such a *system-of-systems* [12, 30] (SoS) will enable leveraging existing high-fidelity simulation models without having to create new ones from scratch. However, this approach is still plagued by several major challenges that all arise from the complexity of chaining together multiple simulations in this way: (C1) *monolithic simulations* that are designed to be used in isolation, (C2) *complex configurations* for each model, (C3) *non-standard data exchange* for passing data between them, and (C4) *long execution times* for each individual simulation that are not amenable to interactive visual analytics.

To address these challenges, we present **VASA** (Visual Analytics for Simulation-based Action), a visual analytics platform for interactive decision making and computational steering of these types of large-scale simulation pipelines based on a visual analytics approach. The VASA Workbench application itself is an interactive desktop application that binds together a configurable pipeline of distributed simulation components. It eables the analyst to visually integrate, explore, and compare the inter-related and cascading effects of systems of systems components and potential final alternative outcomes. This is acheive by visualizing both intermediate and final results from the simulation components using a main spatiotemporal view as well as multiple secondary views. The tool provides an interface for the analyst to navigate in time, including stepping backwards and forwards, playing back an event sequence, jumping to a particular point in time, adding events and threats to the timeline, and initiating mitigation measures. Moreover, it allows them to select between or combine different ensemble outputs from one simulation to be fed to other SoS components and explore consequences. Using this interface, an analyst could for example add a weather event (e.g., either an existing hurricane from a historical database, the union of several ensemble output paths, or simulation of a new one) to a particular time, and then step forward a week to see its impact on roads, the power grid, food distribution, and total economic impact in southern United States.

The simulation components provide the main functionality to the VASA platform. Each simulation component communicates with the Workbench using a representational state transfer (REST) API that standardizes the data and parameter exchange. The data flows and parameters passed in the pipeline can be configured using the Workbench application using a graphical interface. Furthermore, the Workbench also includes a local *simulation proxy* for each remote simulation component that provides real-time approximations of each simulation model to enable using them for interactive visual discourse. This feature also provides the computational steering functionality of the Workbench: after configuring a simulation run in an interactive fashion, the analyst can launch the (possibly lengthy) execution from the Workbench. The Workbench then provides tools to manage the simulation pipeline, for example to prematurely shut down a simulation component to accept a partial result, or to skip a particular run.

Our work on the VASA project has been driven by stakeholders interested in supply chain management of food systems, with an initial working example of a food production to restaurant system. For this reason, other than the VASA Workbench application and the protocols and interfaces making up the platform, we have also created VASA components for simulating weather (including storms, hurricanes, and flooding), the power grid, supply chains, transportation, and food poisoning. We describe these individual components and then present an example of how the VASA platform can be used to explore a what-if scenario involving a major hurricane sweeping North Carolina and knocking out a large portion of the road networks and power grid. We also illustrate how the tool can be used to simulate food contamination outbreaks and how this information can be used to track back the contaminated products to the original distribution centers.

2 Background

Visual analytics [38], can be a powerful mechanism to harness simulation for understanding the world. Below we review the literature in visual analytics for simulation and computational steering, as well as appropriate visual representations for such spatiotemporal data.

2.1 Simulation Models

The potential for applying visual analytics to simulation involves not only efficiently presenting the results of a simulation to the analyst, but also building and validating large-scale and complex simulation models. For example, Matkovic et al. [27, 28] show that visual analytics can reduce the number of simulation runs by enabling users to concentrate on interesting aspects of the data. Maciejewski et al. [23] apply visual analytics techniques to support exploration of spatiotemporal models with kernel density estimation and cumulative summation. This work was extended to a visual environment for epidemic modeling and decision impact evaluation [1]. Similarly, Andrienko et al. [5] propose a comprehensive visual analytics environment that includes interactive visual interfaces for modeling libraries and supports selection, adjustment, and evaluation of such modeling methods. Our work is different from this prior art in that our approach combines multiple components in a simulation pipeline, where each stage in the pipeline produces visualization for analysis.

Supply chain management is also a multi-decisional context where what-if analyses are often conducted to capture provenance and processes of supplies. Simulation is recognized as a great benefit to improve supply chain management, providing analysis and evaluation of operational decisions in the supply process in advance [37]. With the IBM Supply Chain Simulator (SCS) [9] and enterprise resource planning (ERP), IBM is able to visualize and optimize nodes as well as relations in the supply chain [20]. Perez also developed a supply chain model snapshot [31] with Tableau. However, existing visualizations of supply chain are mostly limited to either local supply nodes or a metric model rather than managing the overall supply process.

2.2 Computational Steering

Computational steering refers to provoding user control over running computations, such as simulations. Mulder et al. [29] classify uses of computational steering as model exploration, algorithm experimentation, and performance optimization. Applications include computational fluid dynamics (CFD) [13], program and resource steering systems [40], and high performance computing (HPC) platforms [7].

For all of the above applications, the user interface is a crucial component that interprets user manipulation for reconfiguration of data, algorithms, and parameters. Controlling, configuring, and visualizing such computational steering mechanisms is an active research area. Waser et al. proposed World Lines [41], Nodes on Ropes [42], and Visdom [33] as well as an integrated steering environment [33] to help users to manage *ensemble simulations*—multiple runs of the same or related simulation models with slightly perturbed inputs—of complex scenarios such as flood simulations. In the business domain, Broeksema et al. [8] propose the Decision Exploration Lab (DEL) to help users explore decisions generated from combined textual and visual analysis of decision models rooted in articifical intelligence.

2.3 Spatiotemporal Data

Spatiotemporal visual analytics systems enable users to investigate data features over time using a visual display based on geographic maps [3]. In these systems, color, position, and glyphs display features of different regions by directly overlaying the data on the map.

Many approaches to visual analytics for spatiotemporal data exist. Inspired partly by a survey by Anselin [6], we review the most relevant ones below. Andrienko and Andrienko [4] use value flow maps to visualize variations in spatiotemporal datasets by drawing silhouette graphs on the map to represent the temporal aspect of a data variable. Hadlak et al. [16] visualize attributed hierarchical structures that change over time in a geospatial context. Fuchs and Schumann [15] integrate ThemeRiver [17] and TimeWheel [39] into a map to visualize spatiotemporal data. Ho et al. [18] present a geovisual analytics

framework for large spatiotemporal and multivariate statistical flow data analysis using bidirectional flow arrows coordinated and linked with a choropleth map, histogram, or parallel coordinates plot. Our approach is different from those in that our system provide a visual analytics environment for managing and analyzing the results from multiple types of simulations.

Some approaches enable analysis of spatially-distributed incident data. Maciejewski et al. propose a system for visualizing syndromic hotspots [24, 22] while Malik et al. [25] develop a visualization toolkit utilizing KDE (Kernel Density Estimation) to help police better analyze the geo-coded crime data. The latter system is extended to a visualization system [26]where historic response operations and assessment of potential risks in the maritime environment can be analyzed. In our work we also employ KDE for visualizing spatial distrubution of ill people who consumed contaminated food in a supply chain.

3 Design Space: Steering System-of-System Simulations for Modeling Societal Infrastructure

Computational steering is defined as user intervention in an autonomous process to change its outcome. This approach is commonly utilized in visual analytics [38] to introduce a human analyst into the computation loop for the purpose of creating synergies between the analyst and computational methods. In our work, the autonomous processes we are studying are simulation models (often based on discrete event models) that are chained together into asynchronous simulation pipelines where the output of one or several simulations becomes the input to one or several other simulations. Such a simulation pipeline is also a *system-of-systems* [12, 30] (SoS): multiple heterogeneous systems that are combined into a unified, more complex system whose sum is greater than its constituent parts. Synthesizing all these components yields the concept of visual analytics for *steering system-of-system simulations*: the use of visual interfaces to guide composite simulation pipelines for supporting sensemaking and decisionmaking. In this work, we apply this idea to modeling societal infrastructure, such as transportation, power, computer networks, and supply chains.

In this section, we explore the design space of this concept, including problem domains, users, tasks, and challenges. We then derive preliminary guidelines for designing methods supporting the concept.

3.1 Domain Analysis

A wide array of problem domains may be interested in creating large-scale system-of-system simulation pipelines for studying impacts on societal infrastructure. Our particular domain is for business intelligence for supply chain logistics in the fast-food business, but we see multiple potential applications (each with a specific example):

- **Supply chain logistics:** Impact of large-scale weather events on the distribution of goods (particularly perishables, e.g., food).
- **Public safety:** Crime, riots, and terrorist attacks on critical infrastructure, such as on roads, bridges, or the power grid.
- **Food safety:** Incidence, spread, and causes of food contamination, often due to weather (power outage) or transport delays.
- **Cybersecurity:** Societal impact of cybersecurity attacks, such as on power stations, phone switches, and data centers.

3.2 User Analysis

The intended audience for computational steering of simulation models using visual analytics are what we call "casual experts": users with deep expertise in a particular application domain, such as transportation, supply chain, or homeland security, but with limited knowledge of simulation, data analysis, and statistics. Their specific background depends on the problem domain; for example, they may be business or logistics analysts for supply chain applications, police officers for public safety, and homeland security officials for food safety and cybersecurity. Because of this "casual" approach—a term we borrow from Pousman et al.'s work on casual information visualization [32]—our intended users are motivated by solving concrete problems in their application domain, but are not necessarily interested in configuring complex simulation models and navigating massive simulation results.

Even if our primary user audience is these casual experts, it is very likely that the outcome of a simulation steering analysis will be disseminated to managers, stakeholders, or even the general public [38]. Thus, a secondary user group for consuming our analysis products is laypersons with an even more limited knowledge in mathematics, statistics, and data graphics.

3.3 Task Analysis

Based on our review of the literature (Section 2) as well as feedback from domain experts, we identify a preliminary list of high-level tasks for steering system-of-system simulations for societal infrastructure:

- Increasing *preparedness* for potential scenarios;
- Improving the *resilience* of an organization; and
- Planning for *mitigation and response* to a situation.

3.4 Challenges

Modeling the real world is a tremendously difficult and error-prone process. However, we leave concerns about the fidelity, accuracy, and quality of a simulation to research within the simulation design. Rather, in this subsection we concern ourselves with the challenges intrinsic to connecting multiple individual simulation models into large-scale pipelines. In the context of simulation steering for such pipelines, we identify the following main challenges:

C1 **Monolithic simulations:** While individual high-fidelity simulation models exist for all of the above components and threats, these models are monolithic and not designed to work together.

C2 **Complex relationships:** Each high-fidelity simulation model consists of a plethora of parameters and controls that require expertise and training, which is exacerbated when several such models are combined into a single model.

C3 **Non-standard data:** No standardized data exchange formats exist for passing the output of one simulation model, such as for weather, as input to another model, such as supply chain routing.

C4 **Long execution times:** Most state-of-the-art, high-fidelity simulation models require a non-trivial execution time, often on the order of minutes, if not hours. Such time frames are not amenable for real-time updates and interactive exploration.

C5 **Uncertainty and fidelity:** Chaining together multiple simulations into a pipeline may yield systematically increasing errors as uncertain output from one model is used as input to another. This is compounded by the fact that heterogeneous simulation models may have different levels of fidelity and accuracy.

3.5 Design Guidelines

Based on our review of the problem domain, users, and tasks above, as well as the challenges that these generate, we formulate the following tentative guidelines for designing visual analytics methods for steering system-of-system simulation pipelines:

G1 *Simulations as standardized network services:* Distributing simulation models as network services avoids the trouble of integrating a monolithic design with another system (C1) and automatically provides a data exchange format (C3). The simulations also become decoupled, which means they can be parallelized and/or distributed in the cloud to manage long execution times (C4).

G2 *Simulation proxies for interactive response:* Meaningful sensemaking in pursuit of one of the high-level tasks in Section 3.3 requires real-time response to all interactive queries. This means that long execution times (C4) of simulation models in the pipeline should be hidden from the user. We propose the concept of a *simulation proxy* as an approximation of a remote simulation service that is local and capable of providing real-time response at the cost of reduced (often significantly) accuracy.

G3 *Visual and configurable relationships:* The interactive visual interfaces routinely employed in visual analytics may help to simplify and expose the complex configurations necessary for many high-fidelity simulation models (C2), even for non-expert users.

G4 *Partial and interruptible computational steering:* Once an analyst has configured a simulation run using simulation proxies (G2) and visual mappings (G3), the full simulation pipeline must be invoked to calculate an accurate result. A full-fledged simulation run may take minutes, sometimes hours, to complete. The computational steering mechanisms provided by the software should provide methods for continually returning partial results [14] as well as interrupting a run halfway through.

G5 *Visual representations of both intermediate and final results:* To fully leverage the power of visual analytics, we suggest using interactive visual representations of simulation results. Such visualizations should be used for both intermediate data generated by a simulation component anywhere in the pipeline—which would support partial results and interrupting a run at any time—as well as for the final results. All visual representations should be designed with uncertainty in mind (C6), and providing intermediate visualizations should also help in exposing propagation of increasing error. Finally, it may also be useful to use visual representations for the approximations created by simulation proxies (G2), but these should be clearly indicated as such.

4 VASA: Overview

As previously described, our VASA system is a distributed component-based framework for steering system-of-system simulations for societal infrastructure. Figure 1 gives a conceptual model of the system architecture. At the center of the system is the VASA Workbench (Figure 2), a user-driven desktop tool for configuring, steering, and exploring simulation models, impacts, and courses of action. The workbench provides a visual analytics dashboard based on multiple coordinated views, an event configuration view, and a computational steering view. The workflow of the workbench revolves around initiating, controlling, analyzing, exploring, and handling events from the remote simulation components as well as the local simulation proxies.

Within the dashboard, events are displayed in a selectable calendar view (a) where each event's name, dates and a user-selected representative attribute (e.g., storm's maximum wind speed) are shown. The selected events from (a) are listed based chronologically in the event viewer (b) where a user can select times for investigation. In (b-1), various options are provided, including initiating simulations (e.g., cyberattack, storm simulations, distribution re-routing), selecting combinations of events (union, intersection, difference), selecting event visualization modes (polygons, contours), and chronological playback.

Users can fix a time within an event for comparison (right-clicking on a event's black rectangle) and a red mark is shown in the upper right corner of the associated rectangle(b-2), and the impact is shown in the main geospatial view (d-1). We provide a legend window (c) for selected properties (e.g., distribution centers, restaurants, power plants and other infrastructures) and the geographical view (d) provides the simulation results including event evolution, routing paths, and impacts on critical infrastructures. A food delivery schedule to each store within a supply chain is provided in (e) where the x-axis presents corresponds to different restaurants while the y-axis represents different food processing centers or different types of foods. Here, the darker the red, the larger the quantity of the delivered food. The quantity information is provided in a tooltip that helps a user to estimate possible losses. This view enables traceback analysis (e.g., which type of food was contaminated from which processing centers, how much contaminated food was delivered to which store) for food contamination incidents.

5 VASA: Components

Our current VASA suite consists of four simulation components that implement the VASA interface: components for weather, critical infrastructure, routing, and supply chains. We review each of these next.

5.1 Weather Component

In order to provide clients with a one-stop source for weather data, we implement a server that asynchronously amasses data from various online sources and presents it to clients through a RESTful web interface. This provides access to various data through a singly authenticated service that provides consistent and convenient APIs for data acquired from many sources.

5.1.1 Simulation Model

For example, a collaboration of several research centers runs the ADCIRC model during hurricane season off the east and gulf coasts of the U.S. When storms are present, these models are run every four hours, producing ADCIRC-formatted datasets at fixed intervals forward from the initial times. These results are made publicly available using THREDDS and OPeNDAP for cataloging, discovery and data access. When this data appears, we import it onto a VASA server, and provide a simple RESTful API to access the data in convenient multiresolution formats. Similarly, NOAA produces wind-speed probabilities along the tracks of storms as contours at 34, 50, and 64-knot levels. This data is also imported asynchronously onto the VASA service and provided through the VASA RESTful API.

5.1.2 Simulation Proxy

The proxy in this component has two roles. The first role is to prepare all event data sets from the remote event server. Therefore, the system first checks for new updates from the server. If there is a new update, it retrieves the data and saves it on the local workbench for faster loading. The second role is to visualize new status of an event on the date that a user selected and notify the status change of the event to other proxies. An example status change is a user changing the start date of a hurricane in the event viewer. When this happens, the proxy visualizes a new status of the hurricane on the date and notifies this change to other components, which initiates each proxy's work (e.g., estimating an area without power and impassable roads).

A user can select the hurricane visualization type either as polygons or contours for estimation by clicking a button as shown in Figure 2 (b-2, the last button). In the polygon mode, two probability models (blue with two different opacities) are projected as shown in the magnification view in Figure 2. Here, the smaller polygon means an expected path with high probability, and a larger one presents an expected path with low probability. When a user fixes a hurricane, the hurricane turns red for comparison to other paths (of other hurricanes). For example, in Figure 2 the path of Hurricane Irene on August 24, 2011 is projected (blue) and the path of Hurricane Sandy in October 27, 2012 is presented in red for comparison.

In the contour mode, hurricanes are drawn using three different sizes of contours, each of which represents mean areas in different wind speeds (e.g., Hurricane Irene in our simulation model has 64 knot highest wind speed at the innermost contour, and 34 knot lowest wind speed at the outermost contour as shown in Figure 6). To utilize different wind speeds in simulation steering, a user can set up a threshold for infrastructures (e.g., a power generation unit is disabled if the wind hitting the plant has speed higher than 34 knot). In addition, a user can apply one of the contours for a time. For example, Figure 6 (top-right) presents which power generation units are affected when a contour with 34 knot hits the area. Here red circles represent affected restaurants and red circles present the impacted power generation units supplying electricity to those restaurants.

5.1.3 Implementation Notes and Performance

From the client's point of view, the VASA API consists of URLs that encode procedures and parameters that, when issued, return JSON objects containing the results. This provides a very simple interface for use both by browser-based visualization UIs that use AJAX to issue requests asynchronously, and other native platforms that provide equivalent access through language-specific interfaces.

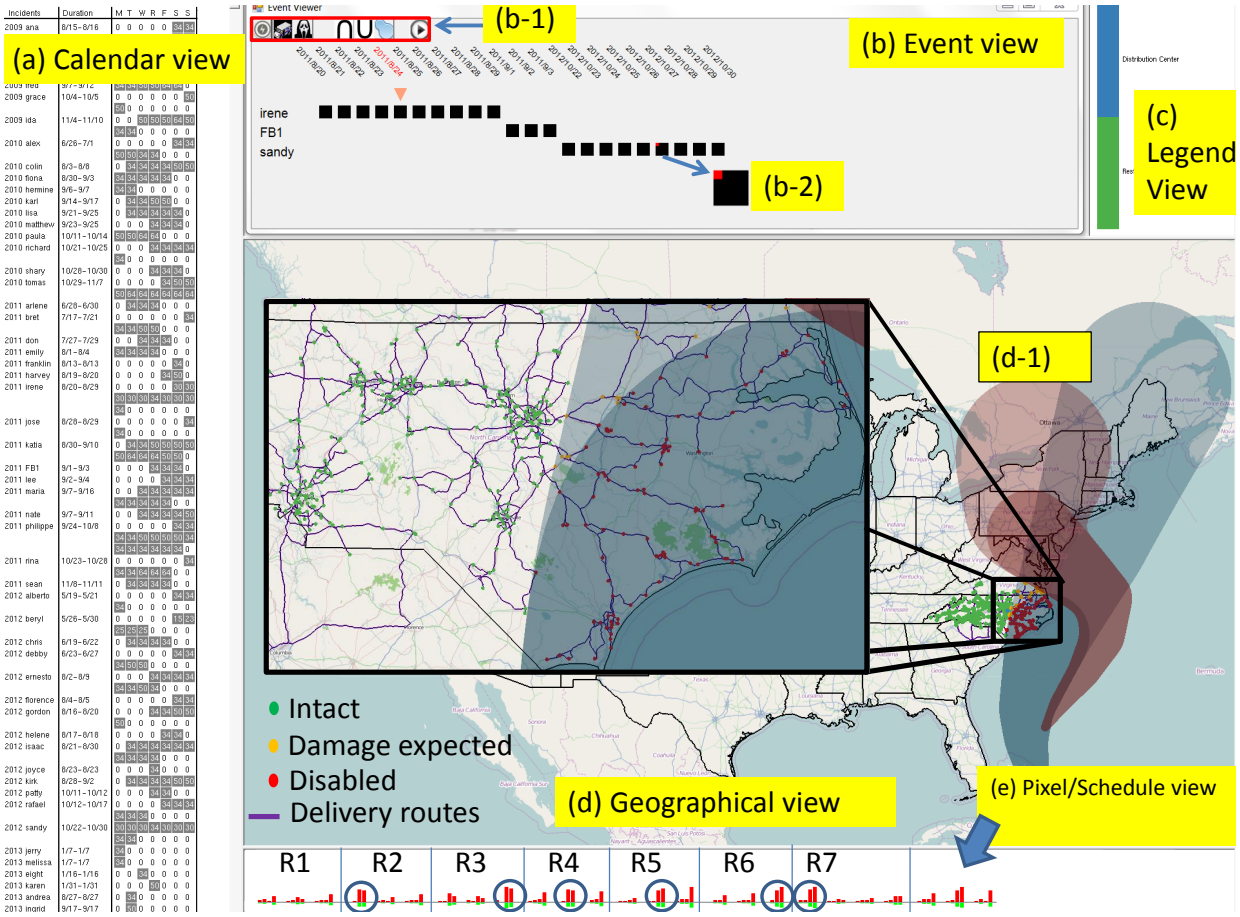


Fig. 2. Multiple coordinated views in the VASA Workbench. (a) Calendar view with available events (e.g., weather, food poisoning, cyberattack, etc). (b) Event timeline for configuring events. (b-1) Event buttons. (b-2) Fixed event. (c) Map legend. (d) Geographical map. (d-1) Hurricane (red). (e) Pixel/Schedule view showing food deliveries. Each area divided by a blue line means a route that visits 3–4 restaruns, 3 time a week. This view also can be used for pixel-based visualization.

5.2 Critical Infrastructure Component

Widespread emergencies such as hurricanes, flooding, or cyberattacks will often affect multiple societal infrastructures. High winds and flooding from a hurricane, for example, could knock out parts of the power grid, the effect of which would cascade to traffic signals, the communications network, the water system, and other infrastructures. The flooding might simultaneously make parts of the road network impassable. These breakdowns would affect critical facilities such as schools, hospitals, and government buildings. For longer-lived disasters, food distribution might break down due to power outage, route disruption, or other cascading effects. The purpose of VASA's critical infrastructure component is to simulate how such external emergencies, modeled in other components, will impact critical infrastructure.

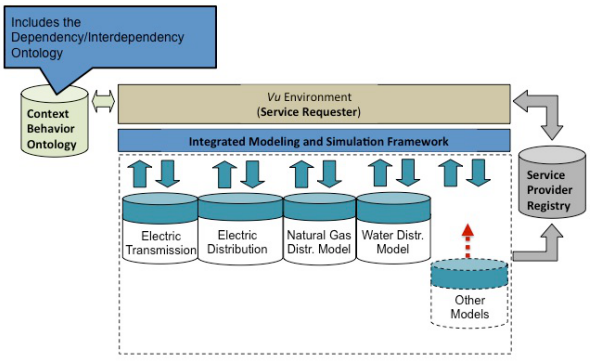


Fig. 3. Vu environment showing the modular structure where different simulation submodels can be inserted.

5.2.1 Simulation Model

To capture these complex, multifarious, and dynamic effects, we developed a simulation model that takes into account the interrelationships between critical infrastructure systems. The simulation is built within the Vu environment (Figure 3), which provides a rule-based framework for integrating multiple infrastructure components at a high level. This results in an interdependency ontology. Thus, for example, a breakdown of a power substation would immediately cascade to power loss at points on its distribution network. If a school were a node in the distribution network, it would be switched to backup power that, after a given time, would also shut down. Likewise, telecommunication nodes would switch to backup power that might also shut down after its prescribed duration. There could also be outages due to power load imbalances at other points in the grid.

These interlaced critical infrastructures are captured in a set of networks, with each node having a set of properties according to its category and the edges providing a dependency rule according to the category and state of the connected nodes. Relations between networks is captured by edges between nodes in the two networks. The timings of interdependencies and state changes are set according to a universal clock, so that any simulation of cascading effects evolves over time and space (since nodes are geographically located). The rules for networks and interdependencies are set in consultation with experts (in the case of the power grid, for example) or through consultation of the appropriate literature for an infrastructure. However, some of the interdependencies are not directly known, even by experts, since measures or simulations linking some infrastructures have never been done or validated. In this case, we define plausible rules that produce outcomes consistent with experience. This is in fact an advantage of the

Vu approach in that it permits investigating interdependencies at a high level with a quick adjustment of the model and turnaround of results. These interdependencies can then be studied in depth as needed.

The Vu approach will not capture detailed interdependency effects, transients, or complex interactions. However, it has been shown to be quite successful as a high-level simulator for a range of applications. To achieve this, it is modular, allowing new modules to be dropped in (Figure 3) and, as soon as the interdependency ontology is set up, simulations to be run. This makes it easy to configure simulations for several different infrastructures and even involving other types of simulation models, such as population and economic models.

5.2.2 Data

Our prototype system currently uses data from the state of North Carolina, and the data collection and organization process involves locating and identifying components of the various infrastructures for the state. We use publicly available data sources, in some cases identifying infrastructure components by indirect means. For example, comprehensive information about the electrical grid is closely held by the utility companies. However, we have shown our results to utility company officials and received confirmation as to their high level accuracy.

The infrastructures we currently model include the electric grid; the communications network including TV stations, radio stations, cellular switch controls, and cell towers; transportation facilities including airports, bus terminals, rail lines and terminals, bridges, tunnels, and ports; the road network including main and secondary roads; natural gas pipelines and pumping stations; critical facilities including fire stations, police stations, schools, hospitals, emergency care facilities, manufacturing locations, government buildings, and hazmat facilities.

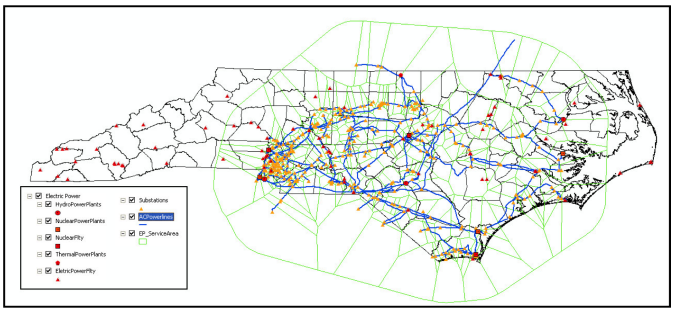


Fig. 4. Power transmission grid with parts of the distribution network.

Figure 4 shows the electric grid, which includes the complete transmission network down to substations for both North and South Carolina. Parts of the distribution network are also included, especially for critical installations. Figure 5 shows the transportation network for North Carolina, including roads, airports, rail lines, etc. For the purposes of VASA, we have also added store and distribution center locations for a large food chain in North and South Carolina. These facilities are linked to the power grid and road networks.

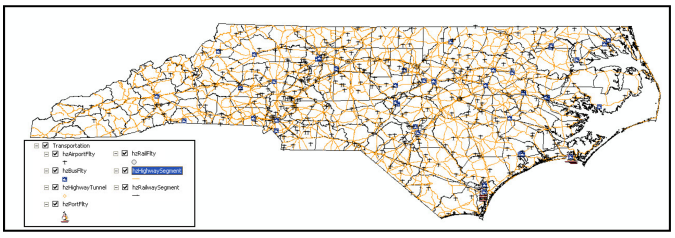


Fig. 5. Transportation network including transportation facilities.

5.2.3 Simulation Proxy

The proxy for the critical infrastructure component maintains a simplified connectivity network of critical infrastructure. In this graph, restaurants are connected to the nearest plant for impact approximation due to an event. When the proxy receives a signal of a new event (e.g., storm path change, new day for approximation), it computes which infrastructures are affected by the event. For example, when a user moves the hurricane simulation forward to a new day, our proxy checks which infrastructures are newly affected and produces an estimate and its corresponding visualization (e.g., condition color changes for restaurants affected by power plant disruptions).

5.2.4 Implementation Notes and Performance

As for all the other VASA simulation components, we use a web service that can accept requests from the VASA Workbench and send either complete or approximate simulation results ready to be presented in the user interface (see Figure 1). The critical infrastructure server itself has two components. One contains a searchable database of the pre-computed ensemble of simulation runs. The other accepts current storm path and other inputs from the weather server, converts them into courses of action, and computes a fresh set of cascading infrastructure disruption results. When a request is issued via the user interface, the Simulation Proxy determines the weather inputs to send to the ensemble database component that immediately selects the closest ensemble simulation for use in the visual analysis. This proxy is then replaced by the more accurate result based on the current weather simulation as soon as it is available. Therefore an emergency response manager can make initial decisions based on the proxy and then refine them, if desired, once the up-to-date result is available.

5.3 Supply Chain Component

Most food systems involve a number of firms from on-farm production of inputs through processing, distribution and retail sales. For the fast food system in VASA, three different firms have collaborated to provide the data on normal system performance: a vertically integrated poultry firm (hatchery to processed chicken), a warehouse and distribution firm, and a fast food restaurant firm. Each firm contributed data from their portion of the supply chain to enable modeling of product movement from farm to restaurant. The type of data provided includes geospatial information on the facilities involved (e.g., feed mills, hatcheries, poultry farms, poultry processing facilities, distribution centers and restaurants), normal transportation routes and scheduled times from each facility to the next facility in the system and details on actual shipment quantities on average (hatchery through processing) or actual shipment records for a limited time frame (distribution centers to restaurants). As an illustration of the amount of data that drives these systems, one week of data on product delivered from the two distribution centers to the nearly five hundred individual restaurants alone is over 120,000 individual records.

Hurricanes pose significant risks for normal supply chain operation from impassable roads, power outages and floodings disrupting facility operation and distribution of products throughout the system. Understanding which routes and locations are likely to be at risk from a storm would enable a firm to develop contingency plans in advance of a storm, thus reducing operational losses immediately after a storm. Given that daily sales at larger fast food restaurants can be $4,300-$7,400, losses can mount up quickly. If in the case of an impending storm or immediate aftermath, near real time rerouting could enable firms to most efficiently maintain their distribution systems for both maintaining product distribution and retrieving of food from restaurants without power to minimize spoilage losses.

5.3.1 Simulation Model

Food contamination can occur both intentionally or as a malicious act at any point in the supply chain and can result in significant public health consequences, from morbidity to mortality. While firms are required to have information one step forward and one step back in their supply chain, they often have difficulty gaining visibility beyond that. By gathering data from each step in the supply chain, it is possible to trace product from farm through to restaurant and from restaurant back to farm. Using data on actual lot sizes from the firms involved, two illustrative contamination scenarios were constructed to illustrate how differently seemingly similar contamination scenarios would transpire. This system also illustrated a common problem of “hidden nodes” in

the system, i.e., facilities that one firm in the system does not realize are part of its supply chain. One of the poultry slaughter and processing facilities ships raw poultry to a further processing facility that then ships the resulting product to the distribution centers. If there were a contamination at the "blind" facility, neither the distribution firm for the restaurant firm would initially know that it was part of their supply chain. A contamination scenario builder is now under development that would enable users to model a wide range of contamination events and see how they would propagate through the supply chain.

Our simulation model can generate food-borne illness data based on an approach similar to the Sydovat [21] system. There are two major components of the model for generating synthetic illness data: temporal and spatial data. A time series is contructed from its individual components (day-of-week, interannual, interseasonal, and remainder) similar to seasonal trend decomposition. To generate the time series of food-borne illnesses for a user-injected restaurant location, the user defines the mean daily count of illnesses along with seasonal and day of week components. If the historical food-borne illnesses data is available then seasonal and day of week components can be randomly selected from this historical data. Spatial locations for temporal data are generated based on the density distribition that approximate the population in that area. Additionally, users can customize the grid size and density distributions.

5.3.2 Simulation Proxy

Our simulation proxy for the supply chain component maintains a low-fidelity representation of the transport network. This is used together with the weather polygons to approximate when a distribution center and store must shut down. For food-poisoning data, this inherently contains spatially-distributed points of ill people simulated based on the simulation model (Section 5.3.1). To visualize the spatial distribution and the hotspots of the poisoned people, the proxy in this component uses a modified variable kernel density estimation technique with varying scales of the parameter of estimation based upon the distance from a patient location to the k_{th} nearest neighbor [36]. The model used for estimating the number of people poisoned is the same model utilized in Maciejewski et al. [1, 22], but we adjust parameters to consider different population densities in different regions.

5.3.3 Implementation Notes and Performance

The supply chain component is built in ArcGIS and Arc Network Modeler so that storm impacts can model solutions accounting for restaurants out of service (power, flooding) and impassable roadways.

5.4 Routing Component

The purpose of the routing component is to provide a mechanism for other VASA components to find appropriate routes from one facility to another given a dynamically changing world model, where roads may become impassable due to weather or other widespread emergencies.

5.4.1 Data Model

We obtained the addresses of two distribution centers and 505 fast-food outlets, as well as the route information that links the centers to the outlets. We geocoded the addresses using the Environmental Systems Research Institute (Esri) ArcGIS 10.2 Server with the Network Analyst extension, and StreetMap Premium for ArcGIS (TomTom North America data) Geodatabase. We then calculate N shortest path routes, where N is the number of routes specified in the input data, using Esri's Network Analyst Route tool and the StreetMap Premium road network. The road network has a long list of attributes used to determine the shortest route, including road class, speed limit, number of lanes, and weight restrictions.

5.4.2 Simulation Model

The input to the routing component is a GeoJSON polygon representing an area impacted by severe weather (such as a hurricane). The component ingests the GeoJSON object as a polygon barrier in the road network. Attributes of the road network are weighted to create a friction surface which iterates through routing options to determine the optimal route. The model does not currently include current traffic conditions or construction activity, but these factors could be added in the future. Each route minimizes the travel time between the distribution center and the first store or between stores. This set of routes represented the baseline scenario—how delivery trucks would travel under normal circumstances. Since delivery trucks can no longer reach outlets covered by the weather barrier, the routing service recomputes the routes with the barrier in place and returns new routes which avoid the outlets and roads covered by the barrier. If the barrier covers a distribution center, no deliveries will be made to outlets serviced by the center. The routes are output as a set of large GeoJSON objects and sent back to the caller.

5.4.3 Simulation Proxy

The main focus of the proxy in the routing component is on approximating the number of routes that will be replaced if a complete simulation result exists. The proxy investigates which nodes in routes are expected to be disabled when there is an event. Then, after the investigation, it builds a polygon by connecting outer-most nodes and visualizes the polygon. This gives awareness to a user that the routes in the polygon are likely to be changed after a complete simulation. A user can initiate the simulation by clicking the "run" button (Figure 9).

5.4.4 Implementation Notes and Performance

The goal was to use as much Commercial Off-The-Shelf (COTS) software as possible when implementing the routing model. The Esri suite of Geographical Information System (GIS) tools is widely used in a variety of industries and provides a robust set of tools and data. Specifically, we used ArcGIS Server 10.2 with the Network Analyst extension. The server provides web-based services through REST endpoints and provides a robust API accessed with HTTPS GET or POST requests. The VASA workbench initiates a request to the routing service by providing a GeoJSON representation of the affected area. The affected area polygon is input to Network Analyst Service to recalculate the route to traverse around the affected area. The response is two large GeoJSON objects containing a list of outlets no longer reachable, incremental travel time between stops, and the new route. Currently, the route processing requires 2-3 minutes to complete; this can be significantly improved when a production server is commissioned.

6 Examples

We showcase the utility of the VASA Workbench and our current simulation components using three examples: the impact of weather on macro-scale supply chains, foodborne illness contamination and spread, and a simplified cyber-attack on the power grid infrastructure.

6.1 Supply Chains in Hurricane Season

Our first example is the potential impact of hurricanes on North Carolina's critical infrastructure, especially our food distribution network, in North Carolina (NC). Our exploration begins by selecting appropriate historial hurricanes for examination using the calendar view as shown in Figure 2, where each hurricane name, duration, and selected summary attribute (e.g., maximum hurricane wind speed) are provided. While we investigate the paths of these historical hurricanes, we see that Irene in 2011 and Sandy in 2012 passed over NC. Because Sandy passed over only a small area in upper NC (Fig 2 (d), red hurricane polygon), we choose to focus on Irene for further investigation.

One interesting date is August 27, 2011 when Irene passed directly over eastern NC, an area with many power generation facilities, as shown in Figure 6 (top-right, purple circles). After we set up the wind tolerance value for these facilities to be 34 knot, our hurricane proxy instantly estimates which restaurants will be impacted based on the relationships between the units and the restaurants and colors the impacted restaurants red. Here, we also initiated a complete simulation for power outages and transportation network damage. Next, a polygon is shown representing an area where restaurants are disabled and which roads are blocked (bottom-left in Figure 6). To efficiently manage distribution, this impact requires the food provider to change its

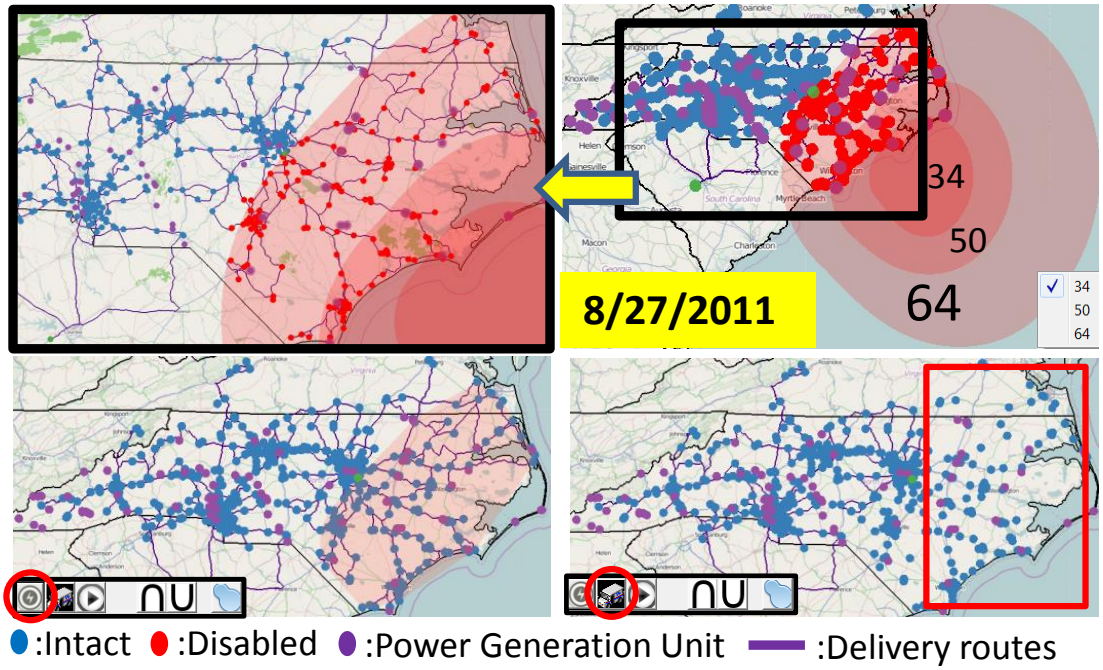


Fig. 6. In this simulation, power generation units were hit by up to 34 knot during Hurricane Irene on August 27, 2011. Our hurricane proxy instantly estimates the impacted restaurants (right-top, left-top). Note that one distribution center (green) is outside the hurricane. After a complete power-grid simulation run is finished (by clicking the circled lightning button), a polygon representing the power outage area is shown. Next, this polygon is sent for use in computing new food delivery paths. Note that food is not delivered to the power outage area (right-bottom, red box).

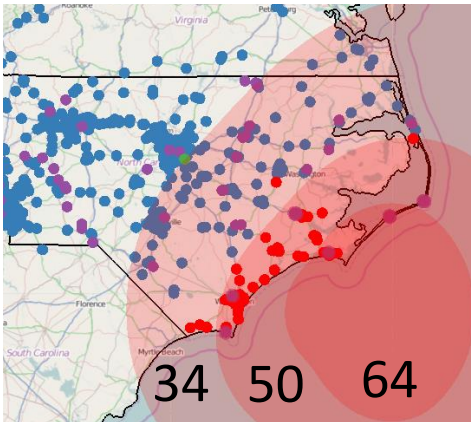


Fig. 7. If the power generation units could have resisted up to 50 knot wind, the number of impacted restaurants will be much smaller.

delivery schedule, and this new routing is computed based on the impacted restaurant polygon and road conditions (e.g., blocked by flooding). After a simulation to compute the new routes (by clicking the truck button in a red circle, right-bottom Figure 6), we see that the updated delivery paths do not include the affected restaurants. The economic loss caused by this event is estimated based on the model in Section 5.3 as being up to $1.13 million. Another possible what-if question is "How different would the result be if the power generation units can resist winds up to 50 knot?" Figure 7 shows the first step of the analysis where we see many fewer restaurants affected compared to Figure 6 top-right (units are resilient to 34 knots). In this case, the estimated losses are less than $333,000.

6.2 Fast Food Contamination

Food poisoning is an illness caused by eating contaminated food containing viruses, bacteria and germ-generated toxins. There are many possible causes of food contamination including storage at inappropriate temperatures [19], improper food handling, and cross-contamination during processing or packaging. As unfortunately experienced several times per year, tracing back the cause of the contamination is a very difficult lengthy process. In this example, we explore a hypothetical scenario demonstrating how VASA can be used to trace-back the root causes of an incident of foodborne illness.

To create the distribution of the ill population, we simulate the distribution of contaminated food to stores, then simulate the illnesses in the neighboring areas using the simulation model discussed in Section 5.3. This creates the common base scenario of reports of people who are ill, their date of illness and their location to create the food contamination scenario for the trace-back investigation.

For example purposes, we simulated these illnesses occurring during a three day span (September 1, 2011 to September 3, 2011) as shown in Figure 8. Since this is almost one week after Hurricane Irene, one may assume that power outages during the storm could be the possible reason behind the contamination. To confirm this hypothesis, we looked at the hot spots in Figure 8 and identified the stores closest to these hot spots. On cross comparison, we can identify the common products/lots in those stores, their distribution center, as well as their delivery mechanisms. As shown in Figure 8 bottom matrices, the rows represent 3 food processing centers and 4 types of food, and there is a column for each restaurant. Each cell is colored such that the darker the red color, the higher the amount of each product provided. Here, the restaurants in the affected area that are selected in the box in the top-left are highlighted with light green boxes. For stores S9 and S12, only one food processing center provided products, while other processing centers supplied most of the food throughout the network. Upon further inspection, one can determining that 3rd and 4th row product lots are common in most of the restaurants where individuals are. Some example routes are shown in Fig. 2 (e) where each route supplies 3-4 restaurants. A red bar means the supplied food and the green bar means the food consumed at a restaurant. Here, we see that a large amount of the third and fourth foods (blue circles in Fig. 2 (e)) are delivered and will all be consumed within a few days. Therefore, these two product lots are good candidates for further inspection in tracing back the contaminated food item.

6.3 Cyberattack on Critical Infrastructure

Part of the mission of the VASA project is to study the impact and mitigation of man-made attacks on societal infrastructure. Cybersecurity is becoming an increasingly important threat to modern society [11]

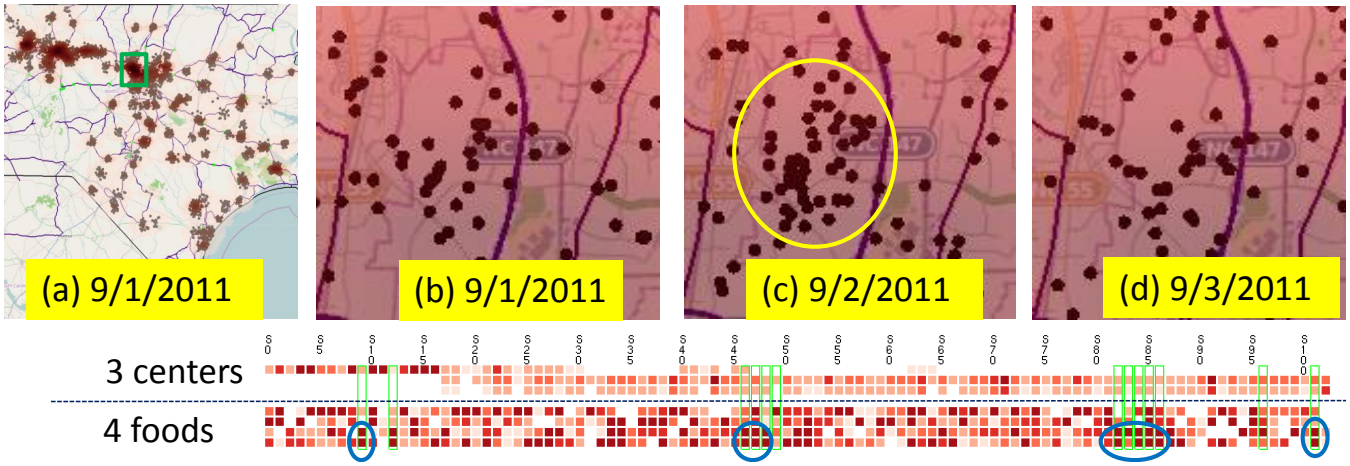


Fig. 8. Ill people caused by contaminated food is presented using a KDE hotspot visualization. In (a), the darker location has a larger number of poisoned people. Brown points mean ill people in the reported location. The locations highlighted by a green box in (a) is magnified in (b), (c) and (d) on different dates. As the timeline shows, the number of ill people increased until 9/2/2011, then started decreasing on 9/3/2011. The bottom matrices show which food processing centers (1–3) were involved and which foods (1–4) were delivered to which store in 8/30/2011, two days before the illness. Here, the restaurants in the light green boxes are the those selected by the thicker green box in (a). We see that a large quantity (darkest red pixels in blue circles) of two foods (third and fourth rows) are commonly provided to restaurants in the area.

Fig. 9. An example of the cyberattack simulation. (left) A user selects to disable power transmission by a cyberattack in the option menu and selects the region shown in the red rectangle. One power plant (purple dot) is included within this rectangle (shown in the dark red circle). (second-left) The infrastructure proxy instantly estimates the affected restaurants (red dots), and a full simulation is initiated. (second-right) Power outage regions are presented by the polygon, and new distribution routes are computed. (right) The new routes are shown as paths and do not include the affected restaurants but, unlike the hurricane scenario, all roads are available for food distribution. For comparison, see the path radiating from the polygon that was not allowed in the hurricane scenario.

and may have a significant effect on an increasingly connected society where power plants and substations are all controlled from afar.

While we do not yet provide a cyberattack module for VASA, many of the simulation components provide direct access to changing the state of particular infrastructure components through VASA. This enables us to simulate a cyberattack by, for example, shutting down a particular or several specific critical infrastructure component even if it is not affected by weather or other natural threats. Figure 9 shows a screenshot of an analyst studying precisely such a scenario where a power plant has been disabled by a cyberterrorist. Here the analyst simulates that the terrorist shuts down the power transmission grid for one of the two power plants in the town by drawing a red rectangle (left) around it. Then, the infrastructure proxy instantly estimates possible affected restaurants and a full simulation is initiated for more accuracy (second-left). The simulation result is presented by a polygon and new route computations can be initiated (second-right). The new routes with impacted restaurants are visualized. Note that this routing example is different from the hurricane case because roads are still passable: purple paths are still shown within the polygon (right).

7 Conclusion and Future Work

We have introduced the notion of visual analytics for simulation steering within the context of societal infrastructure. To our knowledge, ours is the first to study visual analytics for simulation from a *systems-of-systems* [12] perspective, where multiple heterogeneous—often physically distributed—systems are combined into a unified, more complex system in which the linkages between components provide a sum greater than its constituent parts. This notion transcends individual simulation models and instead chains together multiple high-fidelity simulations into large-scale asynchronous pipelines. The VASA system we presented as a practical example of such an approach is a distributed application framework consisting of a central Workbench controlled by an analyst and a set of loosely coupled simulation components implemented as distributed network services.

Big data simulation is a powerful new tool for data science, and while our work on applying visual analytics to this domain is conceptually complete, it really only scratches the surface of what is possible. Future work on the VASA system will involve integrating even more advanced and detailed simulation components, such as high-fidelity power grid models, gas pipelines, and power plants for energy infrastructure; bridges, tunnels, and causeways for transporation networks; and hospitals, police stations, and fire stations for societal infrastructure. In doing so, we envision designing additional novel visual representations and interactions for configuring these components as well as visualizing their proxy, intermediate, and final results.

Acknowledgments

This work was supported in part by the U.S. Department of Homeland Security's VACCINE Center under award no. 2009-ST-061-CI0002.

REFERENCES

[1] S. Afzal, R. Maciejewski, and D. S. Ebert. Visual analytics decision support environment for epidemic modeling and response evaluation. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 191–200, 2011.

[2] C. Anderson. The end of theory: The data deluge makes the scientific method obsolete. *Wired Magazine*, June 2008.

[3] N. Andrienko and G. Andrienko. *Exploratory analysis of spatial and temporal data: A systematic approach*. Springer Verlag, 2006.

[4] N. V. Andrienko and G. L. Andrienko. Interactive visual tools to explore spatio-temporal variation. In *Proceedings of the ACM Conference on Advanced Visual Interfaces*, pages 417–420, 2004.

[5] N. V. Andrienko and G. L. Andrienko. A visual analytics framework for spatio-temporal analysis and modelling. *Data Mining and Knowledge Discovery*, 27(1):55–83, 2013.

[6] L. Anselin. From SpaceStat to CyberGIS: Twenty years of spatial data analysis software. *International Regional Science Review*, 35(2):131–157, 2012.

[7] J. Biddiscombe, J. Soumagne, G. Oger, D. Guibert, and J.-G. Piccinali. Parallel computational steering and analysis for HPC applications using a ParaView interface and the HDF5 DSM virtual file driver. In *Proceedings of the Eurographics Conference on Parallel Graphics and Visualization*, pages 91–100, 2011.

[8] B. Broeksema, T. Baudel, A. G. Telea, and P. Crisafulli. Decision exploration lab: A visual analytics solution for decision management. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):1972–1981, 2013.

[9] S. Buckley and C. An. Supply chain simulation. In *Supply Chain Management on Demand*, pages 17–35. Springer, 2005.

[10] L. Costa, O. Oliveira, G. Travieso, F. Rodrigues, P. Boas, L. Antiqueira, M. Viana, and L. Rocha. Analyzing and modeling real-world phenomena with complex networks: a survey of applications. *Advances in Physics*, 3(60):319–412, 2011.

[11] R. Deibert. Towards a cyber security strategy for global civil society? Technical report, The Canada Centre for Global Security Studies, 2011.

[12] D. DeLaurentis and R. K. Callaway. A system-of-systems perspective for public policy decisions. *Review of Policy Research*, 21(6):829–837, 2004.

[13] H. Doleisch, H. Hauser, M. Gasser, and R. Kosara. Interactive focus+context analysis of large, time-dependent flow simulation data. *Simulation*, 82(12):851–865, 2006.

[14] D. Fisher, I. O. Popov, S. M. Drucker, and m. c. schraefel. Trust me, I'm partially right: incremental visualization lets analysts explore large datasets faster. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 1673–1682, 2012.

[15] G. Fuchs and H. Schumann. Visualizing abstract data on maps. In *Proceedings of the International Conference on Information Visualization*, pages 139–144, 2004.

[16] S. Hadlak, C. Tominski, H.-J. Schulz, and H. Schumann. Visualization of attributed hierarchical structures in a spatiotemporal context. *International Journal of Geographical Information Science*, 24(10):1497–1513, 2010.

[17] S. Havre, B. Hetzler, and L. Nowell. ThemeRiver: visualizing theme changes over time. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 115–124, 2000.

[18] Q. Ho, P. H. Nguyen, T. Åström, and M. Jern. Implementation of a flow map demonstrator for analyzing commuting and migration flow statistics data. *Procedia - Social and Behavioral Sciences*, 21:157–166, 2011.

[19] B. C. Hobbs. *Food poisoning and food hygiene*. Edward Arnold and Co., London, United Kingdom, 1953.

[20] A. Kamran and S. U. Haq. Visualizations and analytics for supply chains. Technical report, IBM, February 2013.

[21] R. Maciejewski, R. Hafen, S. Rudolph, G. Tebbetts, W. Cleveland, S. Grannis, and D. Ebert. Generating synthetic syndromic-surveillance data for evaluating visual-analytics techniques. *Computer Graphics and Applications, IEEE*, 29(3):18–28, May 2009.

[22] R. Maciejewski, R. Hafen, S. Rudolph, G. Tebbetts, W. S. Cleveland, S. J. Grannis, and D. S. Ebert. Generating synthetic syndromic-surveillance data for evaluating visual-analytics techniques. *IEEE Computer Graphics and Applications*, 29(3):18–28, 2009.

[23] R. Maciejewski, S. Rudolph, R. Hafen, A. M. Abusalah, M. Yakout, M. Ouzzani, W. S. Cleveland, S. J. Grannis, and D. S. Ebert. A visual analytics approach to understanding spatiotemporal hotspots. *IEEE Transactions on Visualization and Computer Graphics*, 16(2):205–220, 2010.

[24] R. Maciejewski, S. Rudolph, R. Hafen, A. M. Abusalah, M. Yakout, M. Ouzzani, W. S. Cleveland, S. J. Grannis, M. Wade, and D. S. Ebert. Understanding syndromic hotspots - a visual analytics approach. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technologoy*, pages 35–42, 2008.

[25] A. Malik, R. Maciejewski, T. F. Collins, and D. S. Ebert. Visual analytics law enforcement toolkit. In *Proceedings of the IEEE Conference on Technologies for Homeland Security*, pages 222–228, 2010.

[26] A. Malik, R. Maciejewski, B. Maule, and D. S. Ebert. A visual analytics process for maritime resource allocation and risk assessment. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 221–230, 2011.

[27] K. Matkovic, D. Gracanin, M. Jelovic, A. Ammer, A. Lez, and H. Hauser. Interactive visual analysis of multiple simulation runs using the simulation model view: Understanding and tuning of an electronic unit injector. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1449–1457, 2010.

[28] K. Matkovic, D. Gracanin, M. Jelovic, and Y. Cao. Adaptive interactive multi-resolution computational steering for complex engineering systems. In *Proceedings of the EuroVis Workshop on Visual Analytics*, pages 45–48, 2011.

[29] J. D. Mulder, J. J. van Wijk, and R. van Liere. A survey of computational steering environments. *Future Generation Computer Systems*, 15(1):119–129, 1999.

[30] C. Ncube. On the engineering of systems of systems: key challenges for the requirements engineering community. In *Proceedings of the IEEE Workshop on Requirements Engineering for Systems, Services and Systems-of-Systems*, pages 70–73, 2011.

[31] R. Perez. Supply chain model, April 2011.

[32] Z. Pousman, J. T. Stasko, and M. Mateas. Casual information visualization: Depictions of data in everyday life. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1145–1152, 2007.

[33] H. Ribicic, J. Waser, R. Fuchs, G. Bloschl, and E. Gröller. Visual analysis and steering of flooding simulations. *IEEE Transactions on Visualization and Computer Graphics*, 19(6):1062–1075, 2013.

[34] G. Satell. Why our numbers are always wrong. *Digital Tonto*, October 2012.

[35] G. Satell. Why the future of innovation is simulation. *Forbes*, July 2013.

[36] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, 1986.

[37] S. Terzi and S. Cavalieri. Simulation in the supply chain context: a survey. *Computers in industry*, 53(1):3–16, 2004.

[38] J. J. Thomas and K. A. Cook. *Illuminating the Path: The R&D Agenda for Visual Analytics*. IEEE Press, 2005.

[39] C. Tominski, J. Abello, and H. Schumann. Axes-based visualizations with radial layouts. In *Proceedings of the ACM Symposium on Applied Computing*, pages 1242–1247, 2004.

[40] J. S. Vetter and K. Schwan. Progress: A toolkit for interactive program steering. Technical Report GIT-CC-95-16, Georgia Institute of Technology, 1995.

[41] J. Waser, R. Fuchs, H. Ribicic, B. Schindler, G. Bloschl, and E. Gröller. World lines. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1458–1467, 2010.

[42] J. Waser, H. Ribicic, R. Fuchs, C. Hirsch, B. Schindler, G. Bloschl, and E. Gröller. Nodes on ropes: A comprehensive data and control flow for steering ensemble simulations. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):1872–1881, 2011.

EuroVis Workshop on Visual Analytics (2014)
M. Pohl and J. Roberts (Editors)

Visual Analytics for Risk-based Decision Making, Long-Term Planning, and Assessment Process

Silvia Oliveros-Torres[†1], Yang Yang[†1], Yun Jang[‡2], Ben Maule[3], David Ebert[†1]

[1]Purdue University, USA, [2]Sejong University, South Korea, [3]United States Coast Guard

Abstract

Risk-based decision making is a data-driven process used to gather data about outcomes, analyze different scenarios, and deliver informed decisions to mitigate risk. We describe the design and application of integrated visual analytics techniques and components to support risk-based decision making following a structured risk management process in the US Coast Guard domain. The components proposed perform the following interactive tasks: the identification of risk priority areas, the distribution of pre-computed risk values, and the analysis of coverage versus risk, all of which equip analysts with the tools to examine the different decision factors and assist course of action development in the long-term planning and assessment process.

1. Introduction

Risk-based decision making is a growing operational and business trend that currently lacks interactive tools to aid the decision makers. The term risk is defined as the "potential for an unwanted outcome resulting from an incident, event, or occurrence, as determined by its likelihood and the associated consequences" [Com10]. Therefore, risk-based decision making can be defined as a process that collects and organizes information about different possible outcomes in an ordered structure that helps analysts make informed choices [MMGW04]. Risk-based decision making provides a framework for making decisions and helps identify the greatest risk so the decision maker can prioritize efforts in order to minimize risk and support long-term planning.

However, performing risk analysis and long-term planning is a complex and challenging analytical task, in which the decision maker must set up the problem and determine inputs, outputs, and other factors that might influence the decisions. Research in other areas has shown that individuals often make sub-optimal decisions due to cognitive limitations [SLFE11] and information overload [EM08]. Moreover, the analyst could base his/her decisions on subjective, rather than objective, perception of the risk at hand.

Therefore, we have developed several visual analytics components that can facilitate and improve the process of risk-based decision making. These components, developed through a collaborative user-centered process with the U.S. Coast Guard, use graphical depictions to assist the cognitive process of quantifying and comparing lines of evidence [LCG*12]. Our interactive components facilitate thinking, thereby improving the analyst's understanding of the data and speeding the overall decision making process. The components include feedback and exploratory abilities to examine, filter, and modify certain parameters.

During development, we followed a procedure similar to Sedlmair et al.'s [SMM12] nine-stage framework for conducting design studies. The new components were added to the framework described by Malik et al. [MMME11] because the end users have an understanding and working knowledge of the system.

The new risk-based visual analytics components being applied to visualize and compare risk include the following:

- The use of interactive graphics and choropleth maps to visualize operational risk profiles.
- A method to visualize and identify areas of high risk and compare the changes in risk priority areas over time.
- A method to spatially evaluate and distribute precomputed risk values based on the underlying distribution of cases over time.

† e-mail: {solivero|yang260|ebertd}@purdue.edu
‡ e-mail: jangy@sejong.edu

DOI: 10.2312/eurova.20141137

2. Related Work

In this section, we review previous works that describe the use of visual analytics in communicating risk, some existing models for risk analysis, and different tools to address risk in the maritime security domain.

In risk communication, Lipkus and Hollands [LH99] demonstrated that static images displaying risk characteristics such as risk magnitude and cumulative risk communicate the risk values more effectively than a display of numbers. Savikhin et al. [SME08] demonstrate the benefits of applying visual analytics techniques to aid users in their economic decision making. In contrast, our components provide not only visualizations, but also integrated techniques to analyze the changes of risk values both spatially and temporally.

For risk analysis and modeling, Bonafede and Marmo [BM08] demonstrate that the use of graphs can reduce search times for solutions and for identification of data. They propose four sub-plots with bar graphs and parallel coordinates to compare clients. Feather et al. [FCKM06] describe a risk based decision process with a model that takes into account requirements, risks, and mitigation strategies using bar charts and treemaps. Both papers emphasize that no single visualization technique serves all purposes and instead it is better to use a mix of several. One limitation in their systems is the lack of support of spatiotemporal data. Migut and Worring [MW10] developed a framework that integrates interactive visual exploration with machine learning techniques to support the risk assessment and decision making process. Their visualizations include scatterplots and mosaic plots as tools to build classification models.

Willems et al. [WvdWvW09] presented a geographical visualization using density estimated heatmaps to display vessel movements and support coastal surveillance systems. Pelot et al. [PP08] created a grid colored map representing vessel traffic where they model and identify vulnerable areas. Marven et al. [MCK07] analyzed Search and Rescue operations for the Canadian Coast Guard, exploring the clustering of incident areas with two different models: a Spatial and Temporal Analysis of Crime (STAC) and kernel density estimation (KDE). Abi-Zeid et al. [AZF05] developed SARPlan, a geographic decision support system for planning search and rescue missions, originally developed for aeronautical incidents. Orosz et al. [OSB*10] developed PortSec for decision-making and planning of port resources to address security needs to outside threats and hypothetical scenarios.

3. Visual Analytics in the Risk Management Process

We used the risk management process originally specified in ISO 31000:2009 [ISO09] to provide the initial principles and generic guidelines for risk management. Based on this process, we developed specific goals that our new visual analytics components should achieve:

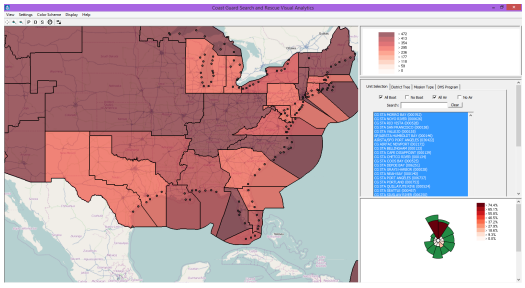


Figure 1: *View of the overall Visual Analytics System*

- Understand areas and missions driving the risk values.
- Identify risk priority areas and how they evolve over time.
- Visualize the geographical distribution of operations.
- Visualize the spatial distribution of the risk.
- Obtain details on demand about the operations.
- Provide a feedback loop if certain parameters change.

Malik et al. [MMME11] focused on the consequences of station closures, but the new additions to the system focus on Risk at the operational level. Such risk is assessed by the USCG Operational Risk Assessment Model (ORAM) [USC12]. Analysts at the Coast Guard Atlantic Area's Operations Analysis Division created this model to support mission planning and analysis of the Coast Guard's mission set. The model combines quantitative and qualitative theoretical frameworks to calculate and compare risk between the eleven Coast Guard statutory missions and geographical areas by providing the Risk Index Numbers (RIN) [USC12]. The RIN is a numerical value that characterizes and quantifies the qualities of risk. RIN values provided include both total risk and residual risk values as shown in Equation 1 [Com10].

$$\text{Total RIN} = \text{Residual RIN} + \text{Mitigated RIN} \quad (1)$$

3.1. Operational Risk Profiles

The first step is to acquire an understanding on how the risk numbers behave for each district as well as how much risk was mitigated. Therefore, there are two main goals in visualizing the Operational Risk Profiles:

- Compare the RIN values between the districts for any given mission or combination of missions.
- Compare the RIN values between missions for any given district.

When performing total versus residual risk analysis, the ratio between the RIN values is more critical than the raw numbers; therefore, we choose a radial layout to focus on ratios and relative values since such layouts inhibit the analysts innate tendency to focus on these numerical details.

We went through several design iterations and presented different alternatives to our end users to gain feedback in terms of which design was the most effective in conveying the information and comparing the distribution of risk. A risk

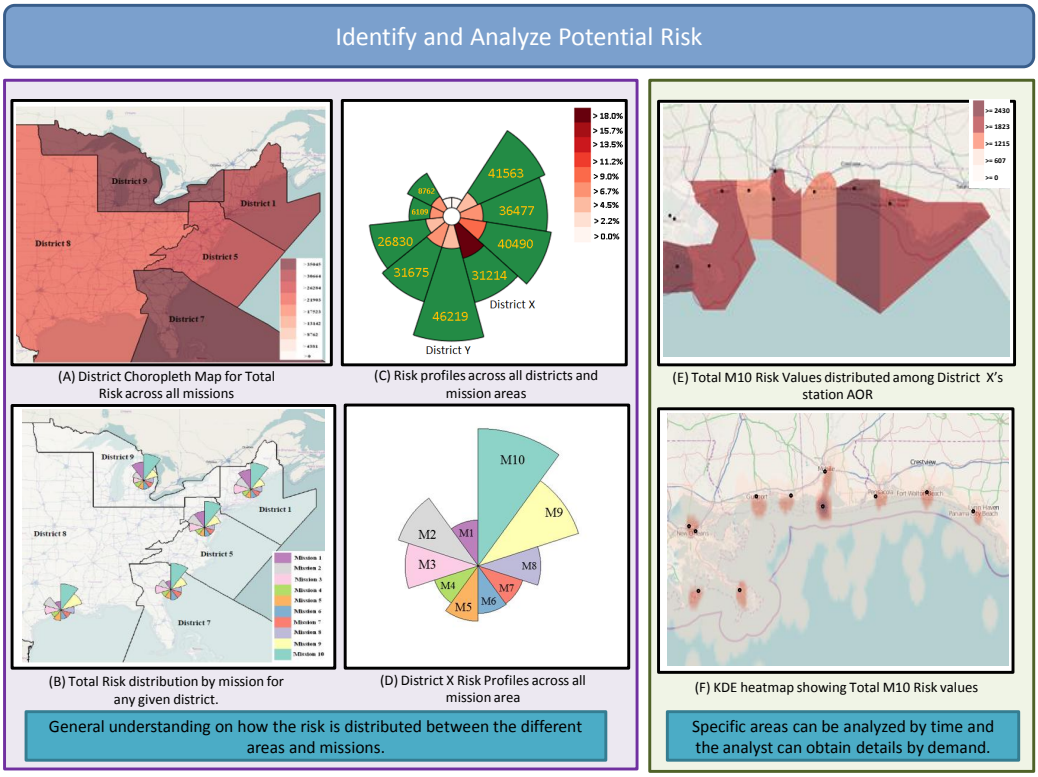


Figure 2: *General process for identifying and analyzing potential risk.*

pie graph was created with eleven fixed pie slices each representing a Coast Guard district as shown in Figure 2-C. The area of each outer pie slice is used to encode the comparison of total risk across districts, with larger pie slice corresponding to higher total risk. The area of inner pie slices represent the comparison of residual risk across districts. Each inner pie slice is also colored on a sequential red scale indicating the ratio of residual versus total risk for a given district. The choice of color (green indicates mitigated risk and red indicates residual risk) is consistent with the Coast Guard's Green-Amber-Red model. We allow interactive filtering by missions to analyze and compare the spatial distribution of risk across districts for any given mission or combination.

3.2. Risk Visualization Using Heatmaps

Next, we need to analyze risk priority areas and how they evolve over time. To quickly identify hotspots, a modified variable kernel density estimation technique (KDE) is employed on the map. Risk at the strategic level is not assigned to a specific unit or station, instead the analyst is able to observe areas with a high density of incidents independent of station location. The heatmap can display the RIN values for total, residual, and mitigated risk. The analyst can switch between the total risk and the residual risk to find hotspots where the risk has not been mitigated and examine the incident details in these zones. Analyzing the incident helps the analyst develop new strategies and courses of action to mitigate the risk.

3.3. Risk Distribution using Choropleth Maps

We utilize choropleth maps in two different ways to help visualize risk. The first option is to visualize any of the Risk values for any given mission or combination of missions by district (Figure 2-A), providing an effective way to present and share the information about risk levels within the U.S.

The second use of choropleth maps (Figure 2-E) highlights the risk distribution of the RIN values per district. During the process it is useful to visualize risk at the station level by using each individual station's Area of Responsibility (AOR). Certain mission's RIN numbers are computed at a district level rather than the station level. Therefore, in order to distribute the RIN values across the stations' AORs, we analyze the underlying incident distributions for a given time period. We use the incidents distribution as a basis to assign risk values across stations given the pre-computed total RIN values by district. The mathematical formula used to compute distributed RIN value for a particular station X that belongs to district Y is:

$$\text{station X RIN} = \frac{\text{Incidents in X}}{\text{Incidents } \forall \text{ stations in Y}} \times \text{district Y RIN} \quad (2)$$

The risk distribution choropleth map provides an easy way to visualize the variations in risk values for individual station's AOR and help identify stations that will potentially require allocation of more resources.

3.4. Visual Analytics System

The overall system provides multiple linked windows and advanced filtering techniques to perform spatio-temporal analysis on the risk data as shown in Figure 1. The system allows the user to visualize historical Coast Guard Data, such as the number and location of incidents that occurred during a certain period of time. It can analyze incidents occurring on specific date ranges to explore seasonal trends and it can filter incidents relevant to the analyst's hypothesis. The addition of the new components enables the Coast Guard analyst to perform risk-based analysis of the operation as well as long term planning by providing new visualization along with feedback loops that control resource allocation.

4. Case Study: Identify and Analyze Potential Risks

To illustrate the use of our system, we present an example use scenario using notional data. In decision making, several questions will drive the analyst in developing the planning strategy: What risks exist in the region and where they are distributed? Where are our resources allocated? What constraints exist in the system that will require a prioritization of resource use?

In a resource constrained environment, we want to use resources in the mission area that provides the greatest return on investment (large amount of total risk but very little residual risk). The first step in the risk management process is to identify potential risks; therefore the analyst begins by looking at the operational risk profile and the district risk choropleth map to observe the risk values at the district level across all mission areas.

Figure 2-C displays the total and residual risk and the ratio between them for all the districts across all mission areas. In this case, we can observe that although District Y has the largest total risk values, it mitigated most of the risk effectively. On the other hand, District X shows less total risk, but the amount of residual risk as well as residual to total risk ratio is the highest as encoded by the darkest red shade. District X can be seen as more problematic than District Y; thus, the analyst will focus more attention on analyzing this particular district. This visualization provides a starting point in understanding how risk is distributed among the different districts and focusing on districts with high risk concentration.

After identifying that District X has the greatest residual risk and the highest risk concentration, the next step is determining the key drivers of risk within a district. This leads the analyst to leverage other components of the risk visual analytics tool to specifically evaluate District X. For instance, the analyst can examine the distribution of risk across different missions in District X as shown in Figure 2-D to identify which mission type has the greatest risk in this district. The analyst can observe that most of the operational risk emerges from one of the missions, in this case M10.

New questions emerge at this stage: Are there several big events that drive the risk, or are there many small events with smaller consequences accumulated to affect the operation? So now we examine the spatial distribution of M10 risk within District X to analyze specific areas of high residual risk. Depending on the data quality regarding spatial location, the analyst has two options for drilling down into specific areas within District X. The first option is to use the risk heatmap described in Section 3.2 to locate risk priority areas, as seen in Figure 2-F. If the spatial location is not available, then we re-distribute the risk to station AORs as described in Section 3.3 and as seen in Figure 2-E.

5. Domain Expert Feedback

The prototype components went through an iterative design refinement process with the collaboration of four Coast Guard personnel: an operation research analyst, a former Coast Guard officer, one in-field officer, and a high level officer. Informal feedback is given below:

"These components aid the analyst in answering the questions that come from developing the planning strategy, often with a speed that was previously unattainable with the Coast Guard's usual brute force processing of thousands of lines of data to calculate summary statistics."

"This system provides a risk informed process for building a defensible planning baseline for the long-term planning process. Understanding the risk profiles provides analytic justification for resource use, and can aid in demonstrating effective application of resource use based on risk."

6. Conclusions

We have demonstrated how our interactive visual analytics components can facilitate the risk management process and evaluate courses of action. Within the maritime context, our interactive visual analytics environment utilizes KDE heatmaps to help identify risk priority areas, multiple designs to visualize risk profiles, a risk distribution choropleth map to visualize the spatial distribution of pre-computed risk values, and the coverage map overlaid with risk distribution for analysis of coverage capability/efficiency as well as potential need for resource reallocation or assets upgrade. Finally, we included a case study that examines the efficiency of Coast Guard operations and provides useful visual reference that can communicate recommendations based on risk management. The described risk-based decision making process serves as a blueprint for future systems dealing with risk values and resource planning.

Acknowledgment

This work was partially funded by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0003. Jang's work was supported in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2013R1A1A1011170).

References

[AZF05] ABI-ZEID I., FROST J. R.: Sarplan: A decision support system for canadian search and rescue operations. *European Journal of Operational Research 162*, 3 (2005), 630 – 653. Decision-Aid to Improve Organizational Performance. 2

[BM08] BONAFEDE C., MARMO R.: Operational Risk Visualization. *Science* (2008), 100–103. 2

[Com10] COMMITTEE R. S.: *DHS Risk Lexicon*. Homeland Security, 2010. 1, 2

[EM08] EPPLER M., MENGIS J.: The concept of information overload - a review of literature from organization science, accounting, marketing, mis, and related disciplines (2004). In *Kommunikationsmanagement im Wandel*, Meckel M., Schmid B., (Eds.). Gabler, 2008, pp. 271–305. 1

[FCKM06] FEATHER M., CORNFORD S., KIPER J., MENZIES T.: Experiences using Visualization Techniques to Present Requirements, Risks to Them, and Options for Risk Mitigation. *2006 First International Workshop on Requirements Engineering Visualization (REV'06 - RE'06 Workshop)* (Aug. 2006), 10–10. 2

[ISO09] ISO 31000, Risk Management Principles and Guidelines, Geneva : International Standards Organisation, 2009. 2

[LCG*12] LINKOV I., CORMIER S., GOLD J., SATTERSTROM F. K., BRIDGES T.: Using our brains to develop better policy. *Risk Analysis 32*, 3 (2012), 374–380. 1

[LH99] LIPKUS I. M., HOLLANDS J. G.: The visual communication of risk. *Journal of the National Cancer Institute. Monographs 27701*, 25 (Jan. 1999), 149–63. 2

[MCK07] MARVEN C., CANESSA R., KELLER P.: Exploratory spatial data analysis to support maritime search and rescue planning. *Geomatics Solutions for Disaster Management* (2007), 271–288. 2

[MMGW04] MACESKER B., MYERS J., GUTHRIE V., WALKER D.: *Quick Reference Guide to Risk Based Decision Making (RBDM): A Step by Step Example of the RBDM Process in the Field*. EQE International, Inc., an ABS Group Company Knoxville, Tennessee, 2004. 1

[MMME11] MALIK A., MACIEJEWSKI R., MAULE B., EBERT D.: A visual analytics process for maritime resource allocation and risk assessment. In *Visual Analytics Science and Technology (VAST), IEEE Conference on* (oct. 2011), pp. 221 –230. 1, 2

[MW10] MIGUT M., WORRING M.: Visual exploration of classification models for risk assessment. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)* (October 2010), pp. 11–18. 2

[OSB*10] OROSZ M., SOUTHWELL C., BARRETT A., CHEN J., IOANNOU P., ABADI A., MAYA I.: Portsec: A port security risk analysis and resource allocation system. In *Technologies for Homeland Security (HST), 2010 IEEE International Conference on* (Nov. 2010), pp. 264 –269. 2

[PP08] PELOT R., PLUMMER L.: Spatial analysis of traffic and risks in the coastal zone. *Journal of Coastal Conservation 11* (2008), 201–207. 2

[SLFE11] SAVIKHIN A., LAM H., FISHER B., EBERT D.: An experimental study of financial portfolio selection with visual analytics for decision support. In *System Sciences (HICSS), 44th Hawaii International Conference on* (2011), IEEE, pp. 1–10. 1

[SME08] SAVIKHIN A., MACIEJEWSKI R., EBERT D.: Applied visual analytics for economic decision-making. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)* (October 2008), pp. 107–114. 2

[SMM12] SEDLMAIR M., MEYER M., MUNZNER T.: Design Study Methodology: Reflections from the Trenches and the Stacks. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis) 18*, 12 (2012), 2431–2440. 1

[USC12] USCG: *Joint, CG Atlantic Area and CG Pacific Area Operational Risk Assessment Model (ORAM), Executive Summary*. Unpublished Technical Document, 2012. 2

[WvdWvW09] WILLEMS N., VAN DE WETERING H., VAN WIJK J. J.: Visualization of vessel movements. *Comput. Graph. Forum 28*, 3 (2009), 959–966. 2

Visual Analytics of User Influence and Location-Based Social Networks

Jiawei Zhang, Junghoon Chae, Shehzad Afzal, Abish Malik, Dennis Thom, Yun Jang, Thomas Ertl, Sorin Adam Matei, and David S. Ebert

1 Introduction

Social media have transformed the way people express opinions, react to evolving and emergent events, and share their whereabouts. When an event occurs, information generated by users who witness or engage in it can provide first-hand accounts and updates. This information is propagated in the online social networks and triggers reactions from other users. Identifying influential users, monitoring the interaction between users, and analyzing information diffusion in social media can improve situational awareness in a crisis situation, and provide significant and reliable information for emergency management. Yet, inferring actionable information from raw social media data is not straightforward. The large volume of data and their multiple dimensions makes this process extremely difficult. The real time streaming nature of social media data introduces additional challenges. Reliance only on fully automated methods with minimum human intervention is not suitable while working with such datasets. Besides that, analysts working on such problems often need to look into the contextual evidence that could help them accept or reject certain hypothesis. Such contextual information could be provided only if application framework supports

J. Zhang (✉) • J. Chae • S. Afzal • A. Malik • D.S. Ebert
School of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana, USA
e-mail: zhan1486@purdue.edu

D. Thom • T. Ertl
Department of Computer Science, University of Stuttgart, Stuttgart, Germany

Y. Jang
Department of Computer Engineering, Sejong University, Seoul, South Korea

S.A. Matei
Brian Lamb School of Communication, Purdue University, West Lafayette, IN, USA

S.A. Matei et al. (eds.), *Transparency in Social Media*,
Computational Social Sciences, DOI 10.1007/978-3-319-18552-1_12

interactive exploration, querying, and visual feedback. In this chapter, we introduce our suite of interlinked visual analytics tools that attempt to overcome these issues.

In the process, we also discuss some recent advances to support space-time indexed data. Such spatiotemporal data have immense value for increasing situational awareness of local events, providing insights for investigations regarding the extent of incidents, their severity and consequences, as well as public behavior during crises. However, the large volume of the data hinders effective exploration and examination, while its volume is significantly and constantly increasing. Also, a relatively small volume of critical data may easily be obscured by the large amounts of data generated every day. Thus, analysts need new methods for dealing with the data volume and its dynamic nature, as well as identifying abnormal events and topics within the data.

Typically, such challenges take advantage of automated data mining techniques, which are utilized to deal with large and/or high-dimensional data sets. Data mining is commonly defined as the process of discovering useful high-level knowledge from low-level data known as Knowledge Discovery in Database (KDD) (Fayyad, Piatetsky-Shapiro, & Smith, 1996). In the data mining process, data exploration is an important step to gain insight into data and create hypotheses. Adding a human intelligence touch to data discovery increases the relevance and validity to data streams. The human knowledge and the ability to quickly understand human behavior play an important role in the data exploration process. Visual analytics aims at integrating the human knowledge and the perceptual abilities of the human mind with automatic data mining procedures (Keim, 2002). Shneiderman (2002) notes that combining human cognitions supported by visualization and automated data mining can lead to novel discovery strategies that preserve user control and enable a more effective data exploration. In this chapter, we describe our visual analytics approaches that allow users to directly manipulate the algorithms in order to understand the workings and results of the data mining algorithms; thereby, enhancing their ability to contextualize data. This serves not only the human analysts but also improves the data mining methodologies. In other words, our interactive visualizations and visual analytic systems can help facilitate knowledge discovery from social media data by enabling analysts to generate, test, and refine their hypotheses.

2 Related Work

In what follows, we primarily consider the case of location-based information for discussing our broader strategy of data mining, discovery and visualization. Location-sensitive data harvested from social networks have become a popular and influential data source for many applications. However, the large volume of data and the unstructured nature of the information hinders exploration and examination. Thus, scalable computational analysis for improving spatiotemporal situational awareness and discovering of critical information within the data are vital research topics and application domains. The following subsections present previous works that have focused on LBSN analysis and the manner in which they have contributed to our own vision.

2.1 Visualization of Social Networks

In order to explore and examine the large number of nodes and links in social networks, some previous studies combine data mining algorithms and visualization techniques (Sun et al., 2009; Yang, Asur, Parthasarathy, & Mehta, 2008). Correa, Crnovrsanin, and Ma (2012) propose an analytical mechanism for measuring sensitivities of nodes in a network using eigenvector and Markov centralities to find important and influential nodes. Crnovrsanin, Liao, Wu, and Ma (2011) demonstrate a visualization system for supporting effective navigation of social networks. The system suggests directions based on the importance of the nodes in the network and past user interactions. In user activities in Twitter, retweeting and replying are important and distinguishable attributes associated with the links between Twitter users compared to traditional social network relationships. Also, these are the key mechanisms for information propagation and transfer in the social networks. There are some studies that focus on the mechanisms of information transfer, although those do not discuss visualization aspects. Suh, Hong, Pirolli, and Chi (2010) study the factors affecting the retweet ability of tweets. They focus on content features (e.g., URLs and hashtags) and contextual features (e.g., the number of followers and followees) to estimate the factors that are significantly associated with retweeting. Macskassy and Michelson (2011) focus on information diffusion behaviors underlying processes by which they decide to retweet. Ho, Li, and Lin (2011) study how information is propagated in micro-blog networks with respect to the number of users influenced, the speed of propagation, and the geographical distance of the propagation.

2.2 Location-Based Social Networks Analysis

As social media platforms move towards LBSN researchers have proposed various approaches to analyze spatiotemporal document collections, in general, and spatio-temporal social media data, in particular. VisGets (Dork, Carpendale, Collins, & Williamson, 2008) provides linked visual filters for the space, time and tag dimensions to allow the exploration of data sets. The user is guided by weighted brushing and linking, which denotes the co-occurrences of attributes. Further works demonstrate the value of visualizing and analyzing the spatial context information of microblogs for social network users (Field & O'Brien, 2010) or third parties like crime investigators (Roth & White, 2010) and urban planners (Wakamiya, Lee, & Sumiya, 2011). Andrienko et al. (2013) describe a visual analysis approach for exploring tweet text and spatiotemporal patterns. Krueger, Thom, and Ertl (2014) extract frequent visited places from vehicle movement data and further use semantics distilled from the social network to decode daily activities of people. MacEachren et al. (2011) demonstrates a visual analytics system to represent tweet density of actual or textually inferred locations. Their work also demonstrates

that social media can be a potential source for crisis management. Bosch et al. (2011) provides a scalable system enabling analysts to work on quantitative findings within a large set of tweets with geo-location. Chae et al. (2012) propose a combination of LDA and Seasonal-Trend Decomposition for abnormal event detection. Researchers also present analysis of LBSN for disaster management and evacuation planning (Chae et al., 2014; Sakaki, Okazaki, & Matsuo, 2010; Terpstra, Stronkman, de Vries, & Paradies, 2012). Ying, Lee, Ye, Chen, and Tseng (2011) present various location-based metrics using spatial information of these LBSNs to observe popular people who receive more attention and relationships within the network.

3 User Influence-Based Dynamic Social Networks

Social media can be utilized as a publicly available data source to identify and gather information pertaining to events of interest. In such scenarios, identifying both individuals of interest (e.g., witnesses), and the potential information they disseminate through their social media networks, can be especially instrumental for decision makers and emergency managers. When a specific event occurs, four types of actors are mainly involved in the social networks: (1) users who engage in the event and post messages, (2) common users who participate in the propagation and forward messages, (3) popular users (including celebrities, opinion leaders, news broadcasters) who accelerate the propagation of messages, and (4) passive users who receive but do not forward the messages (Romero, Galuba, Asur, & Huberman, 2011). Typically, a message containing important information is posted by a witness, diffused through popular or common users, and finally ends with passive users. In this process, two types of influential users are of particular interest: witnesses and popular users. Witnesses provide first-hand updates of information that can help understand and respond to events as quickly as possible. However, in most cases, they are hidden in the massive noise of the crowd. For popular users, there is a delay in the messages to reach them. However, they stand out in the information diffusion process and can provide clues for tracing the source of the messages. Based on the above observations, we develop a visual analytics framework for user networks and information diffusion processes to identify these types of influential users.

3.1 Explicit Connections: Replies and Retweets

Explicit connections among Twitter users mainly include reply/retweet and follower/friend. Reply/retweet connections are generated when users explicitly establish connections with each other. Reply/retweet serve as the most popular way for Twitter users to share and deliver instant messages and can indicate strong

Table 1 Examples of retweet and reply

Type	User	Twitter messages
Retweet	@stoobush	1. RT @rtv6: #BREAKING: shooting reported at Purdue Electrical Engineering building, campus police confirm
Reply	@Hmother8	2. @calmoza shooting on campus!

relationships among them. In contrast, follower/friend connections are formed when users want to get real-time updates of the individuals they choose to follow. Since users can establish or disconnect a follower/friend relationship with few limitations, follower/friend based networks are often large-scale, show little change over short periods of time, and lack any semantic content. These are therefore of less importance than reply/retweet connections when collecting information pertaining to specific events. Consequently, we focus on reply/retweet connections in our work. For a typical retweet/reply message, we define the direction of information diffusion. We illustrate this directionality using two example messages shown in Table 1. Here, message (1) is a retweet message, where the original message is posted by the user @rtv6, and is then retweeted by the user @stoobush. Message (2) is a reply message, where the message is diffused by the user who initiates the conversation (@Hmother8) to the user who receives the message (@calmoza). The network generated based on reply/retweets can thus be considered as directed networks.

3.2 Visualization of Dynamic Networks

Dynamic Twitter networks are rich in valuable latent patterns dealing with multiple information entities, such as Twitter users and topics. Our visualization utilizes user connections in Twitter as the backbone of the dynamic network, and shows the evolution of retweet/reply communications and semantic correlations among the Twitter users. Twitter user relationships are depicted with an overview-based node-link visualization in our system. As shown in Fig. 1 the relationships have three main features:

Nodes: The nodes encode users and the size of each node depicts tweet volume of the user for the currently selected time frame.

Edges: Edges depict user connections. Users with retweet or reply communications are shown using arrow-styled edges to show the direction of tweet propagation. For the retweet communication, the node at the thicker end represents the user who posts the tweet, and the node at the thinner end represents the user who retransmits the provider's tweet. For the reply communication, the node at the thinner end represents the user who posts the tweet, and the user at the thicker end represents the one who replies to the tweet. The system allows users to perform filtering to show either retweet or reply relationships through a set of checkboxes in the interface.

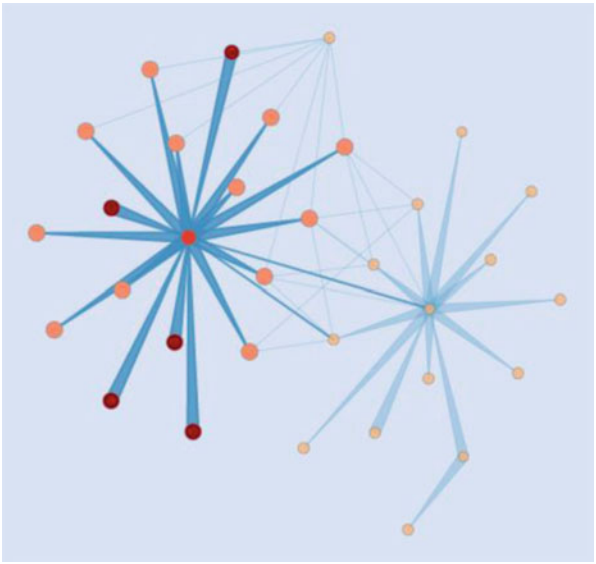


Fig. 1 Forced-directed layout with DOI based visualization

Force-directed layout (Fruchterman & Reingold, 1991): We utilize a Force-directed layout to project users on the screen based on their relationship as shown in Fig. 1. Nodes in the graphs are dynamically changing, since some users may leave, and others may join in. In order to avoid visual confusion caused by user changes, and allow analysts to perceive how users evolve over time, we provide smooth animations to visualize user evolution in different communities and topics. Aside from the above-mentioned overview-based approach, we provide a degree of interest (DOI) based visualization (Card & Nation, 2002) that shows the information propagation process related to a single user. As shown in Fig. 1, when the analyst selects a user, nodes connected to the user are highlighted to present the information diffusion pattern. Users with darker colors serve as sources of information for the users with lighter colors along the propagation path. A combination of the overview-based and DOI-based visualizations allows analysts to iteratively examine information propagation, identify influential nodes, and observe evolution of user networks.

3.3 Interaction Design

Our target end-users include decision makers for emergency and natural disasters, and public safety/law enforcement personnel. Based on close cooperation with these end-users, we base our interaction design and interactive visual tools on supporting simple, intuitive and reversible operations following Norman's Principle of Naturalness (Norman, 1993). We use an interactive details-on-demand (Shneiderman, 1996) ContentLens (Thom, Bosch, Koch, Woerner, & Ertl, 2012)

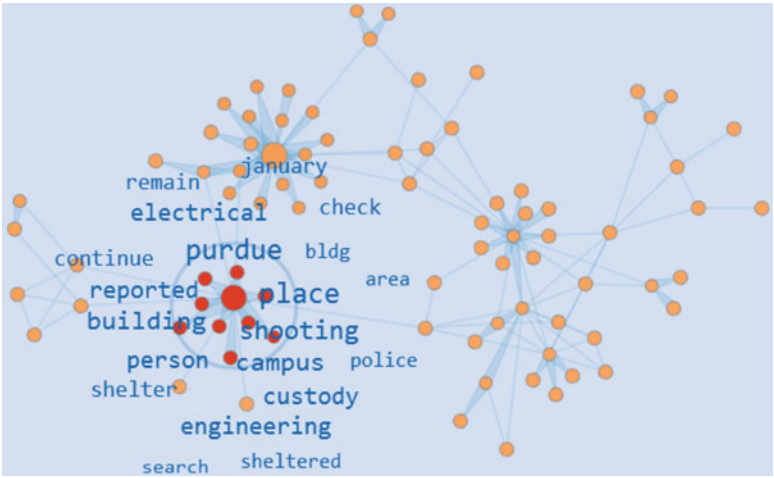


Fig. 2 ContentLens showing major keywords discussed in underlying network

that allows analysts to explore and monitor topics within the social networks. This feature provides analysts with extracted major keywords for their selected level of aggregation, from an individual user to small or large groups within the network as shown in Fig. 2. As the analysts move the ContentLens over the networks, they can focus on specific nodes within the networks. The analysts can dynamically change the diameter of the lens to either overview a large aggregation of tweets or focus on tweets of specific users. By utilizing ContentLens, the analysts are allowed to interactively investigate user activity in their identified social network.

4 Visualization of Location-Based Social Networks for Abnormal Event Detection

Social networks allow people to create a large volume of time-stamped and geo-tagged tweet messages. This requires a tool to cope with large amounts of messages in order to help analysts to explore and detect important messages. Our visual analytics system (shown in Fig. 3) allows analysts to select an initial spatiotemporal context of tweet messages to be represented in the visualization and to serve as a basis for analysis. The spatiotemporal distribution of messages can provide an initial insight to the analysts that can be relevant for their analysis tasks. In the subsequent step, the analysts start with the topic extraction on the analysis context using a data mining model, Latent Dirichlet Allocation (LDA) (Blei, Ng, & Jordan, 2003), that extracts and probabilistically ranks major topics contained in textual parts of the tweets. Users can adjust the configuration parameters of LDA to interactively explore available topics by generalization and specialization.

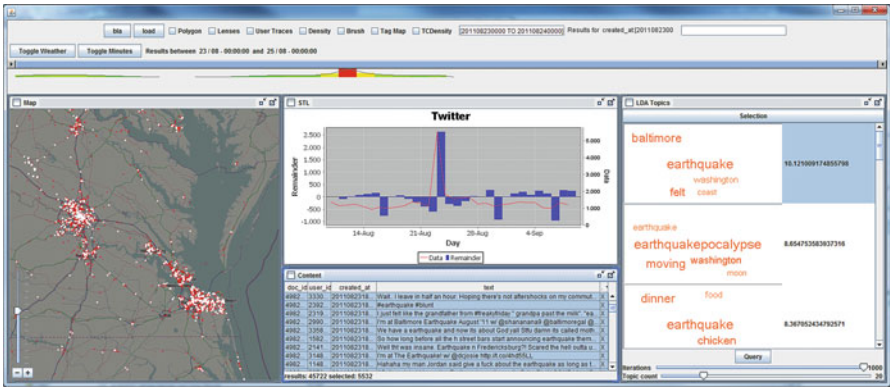


Fig. 3 Social media analysis system including message plots on a map (1), an abnormality estimation chart (2), a tweet content table (3), and a topic exploration view (4). It can be seen how the Virginia earthquake on August 23rd, 2011 is examined using the system. The system detects the earthquake event using our STL based anomaly detection model

The extracted topics can be evaluated and ordered based either on volume-based importance or abnormality estimates computed using the Seasonal Trend Decomposition procedure based on Loess smoothing (STL) (Cleveland, Cleveland, McRae, & Terpenning, 1990). In order to obtain a ranking suitable for abnormal topics analysis tasks, we discard daily chatter by employing STL. This combination of utilizing automatic data mining algorithms that are further facilitated through interactive visualizations enables analysts to discover emerging and abnormal topics from the noise. Additionally, we note that our system also provides the ability to send out email alerts to analysts when a threshold for certain keywords is met (e.g., when the number of twitter messages containing user specified keywords for a certain time step exceeds N). This further assists end-users to detect an emerging situation using social media as an information source.

In Sect. 4.1 we first describe how we utilize the LDA topic modeling to extract the inherent topics from a set of tweet messages. In Sect. 4.2, we explain how we estimate abnormalities for each given topic and re-rank the topics based on the abnormality scores to identify unusual and unexpected topics using STL.

4.1 Topic Extraction

Often when an unusual situation or an unexpected event occurs within an area and a time window, a certain number of tweet messages are generated by the community from the communication of the event. This set of messages implicitly constitutes multiple topics. In order to extract each of the topics exhibited within the collection of messages, we employ the LDA topic model which is a probabilistic topic model that can help organize, understand, and summarize vast amounts of documents.

Table 2 An example of extracted topics and their proportions (Chae et al., 2012)

Rank	Proportion	Topics
1	0.10004	Day back school today
2	0.09717	Ils bout dat wit
3	0.09443	People make hate wanna
4	0.08226	Earthquake thought house shaking
5	0.05869	Earthquake felt quake Washington

The LDA topic model extracts topics from tweets generated on August 23, 2011 around Virginia, where an earthquake occurred on this day. Topics consisting of ordinary and unspecific words have high proportion values, while the earthquake related topics have a relatively low proportion value

The LDA topic model, as presented by Blei et al. (2003), is an unsupervised machine learning technique to identify latent topics from a large document collection. Basically, it uses a "bag of words" approach and assumes that a document exhibits multiple topics distributed over words with a Dirichlet prior. In other words, the LDA assumes the following generative process for each document: First, choose a distribution over topics, choose a topic from the distribution for each word, and choose a word associated with the chosen topic. Based on this assumption, one can apply a Bayesian inference algorithm to retrieve the topic structure of the message set together with each topic's statistical proportion and a list of keywords prominent within the topic's messages. Table 2 shows an example set of extracted topics resulting from the application of LDA to tweets. The topics are ordered by the proportion ranking. The example tweets were generated on August 23, 2011 around the Virginia area. On this day, the area was struck by an earthquake with a magnitude of 5.88. As seen in the table, the last two topics indicate the earthquake event. Figure 4 shows the topic exploration view of the entire system in Fig. 3. We can see most of the topics in the view represent the earthquake event. In our system, the MALLET toolkit (McCallum, 2002) is used for the topic modeling. Prior to the topic modeling, the stemming algorithm KSTEM by Krovetz (1993) is applied to every term in the messages.

4.2 Abnormality Estimation

In order to prioritize the topics extracted using the LDA topic model and allow analysts to discover abnormal events from twitter messages, we employ the Seasonal-Trend Decomposition based on locally-weighted regression (Loess) known as STL (Cleveland et al., 1990) method. We define abnormal events as events that do not occur frequently and regularly. Also, abnormal events generally cover only a small fraction of the social media stream. For example, Table 2 shows the first and second ranked topics consist of ordinary and unspecific words even during an earthquake. The fourth and fifth ranked topics include words indicating the earthquake event of August 2011. From this observation in the

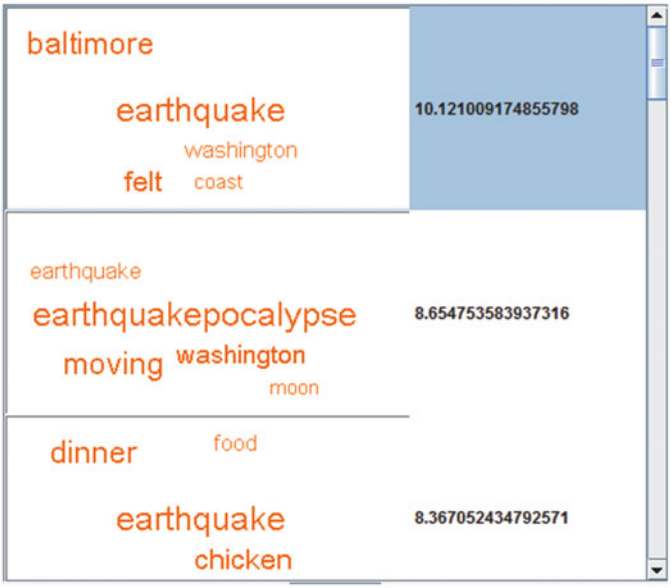


Fig. 4 Topic exploration view. The values in the right column of the view shows the z-scores of each topics. The high scores of the three topics show their strong abnormality

proportions of ordinary and unusual topics over the social media data, we need to differentiate the unusual topics from the large number of rather mundane topics. We utilize STL to identify the topics indicating unexpected and unusual situations. An abnormal event is associated with a set of tweets that provides its contents, location, and time-stamp. To detect abnormal events for a given area and a time window, users select a subset of tweets within the spatiotemporal filter. The LDA topic modeling (described in Sect. 4.1) then extracts a set of topics from the selected tweets. For each topic, we search for relevant tweets in the selected area and time period and a predefined time span of historic data preceding (e.g., 1 month). Tweets are considered relevant if they contain at least one word. A daily tweet count time series is generated from the timestamps of the tweet. The time series can be considered as the sum of three components: a trend component, a seasonal component, and a remainder:

$$Y = T + S + R \tag{1}$$

Here Y is the original time series of interest, T is the trend component, S is the seasonal component, and R is the remainder component. STL works as an iterative nonparametric regression procedure using a series of Loess smoothers (Cleveland, 1979). The iterative algorithm progressively refines and improves the estimates of the trend and the seasonal components. The resulting estimates of both components are then used to compute the remainder: $R = Y - T - S$. Under normal

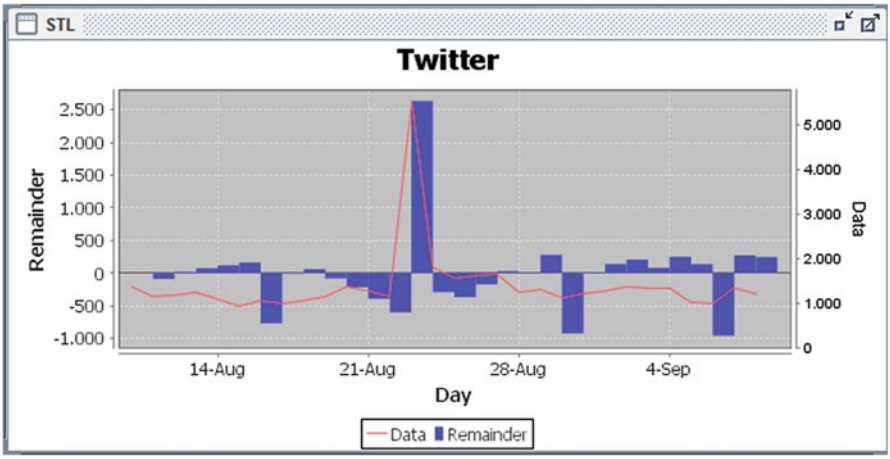


Fig. 5 Abnormality estimation chart. It shows the original time series (*red line*) and remainder values (*bar*) of the first topic in the topic view in Fig. 4. The abnormality degree is extremely high on August 23rd, 2011 (times are given in UTC)

conditions, the remainder will be identically distributed Gaussian white noise, while a large value of R indicates substantial variation in the time series. Thus, we can utilize the remainder values for control chart methods detecting anomalous outliers within the topic time series. We use a 7 day moving average of the remainder values to calculate the z-scores, $z = (R(d) - mean)/std$, where $R(d)$ is the remainder value of day d, mean is the mean remainder value for the last 7 days, and std is the standard deviation of the remainders, with respect to each topic. If the z-score is higher than 2, events can be considered as abnormal within a 95 % confidence interval. The calculated z-scores are thus used as abnormality rating and the retrieved topics will be ranked in the analytics environment according to this estimate. In Fig. 4, the values in the right column of the view shows the z-scores of each topic. The high values of the three topics that related to the earthquake show their strong abnormality. Figure 5 shows the original time (red line) series and remainder component values (bar) of the first topic in the topic view in Fig. 4. We can see the abnormality degree is extremely high on August 23rd, 2011.

5 Case Study

In this section, we demonstrate how our system can be used using historical data from a real-world scenario. On January 21st, 2014, a shooting event occurred around 12:10 PM at Purdue University in West Lafayette, Indiana, USA. In order to explore the evolution of the event and response from Twitter users, law enforcement analysts can utilize our system in order to explore Twitter data generated in real time. As shown in Fig. 6, our system extracts major topics from the tweets and

Fig. 6 Topics from the tweets generated within the Purdue University area during 1 h after the shooting accident on January 21st, 2014

orders the topics based on their abnormality scores. The high abnormal (more than 2) topics are highlighted in red color and the others are black. Our system identifies the shooting incident as an abnormal event and highlights it by assigning it a high abnormality score. In addition, the system generates an email alert to the intended law enforcement recipients as the threshold for law enforcement sensitive keywords is met.

Law enforcement analysts can interactively identify popular users and witnesses for the event. They can enter several keywords that roughly describe the event (e.g., Purdue, shooting, victim, suspect, police, murder). The system then filters the tweets based on the entered keywords and generates user networks. At 12:17:35, the first relevant tweet appears: '*Shooting on campus*?' Over time, more people get involved in the conversation and post messages in their social networks related to the shooting event. Our system also allows analysts to obtain an overview of the evolving topics. The analysts can utilize the ContentLens feature of our system to examine topics over the user group. This is shown in Fig. 2, where the terms shooting, electrical, and engineering are seen as dominant terms. This indicates that the location of the event is around the Electrical Engineering department. As the conversation networks grow, users with a large number of connections with other users join the conversation. In Fig. 7 (Left), the law enforcement analysts select the user *JConline* (a local news media company) to visualize the information propagation patterns from this news source. Five users in dark colors serve as the source of information for *JConline*. We then select these five nodes to explore their information propagation patterns. From 12:30:00 to 13:00:00, these users actively post messages, most of

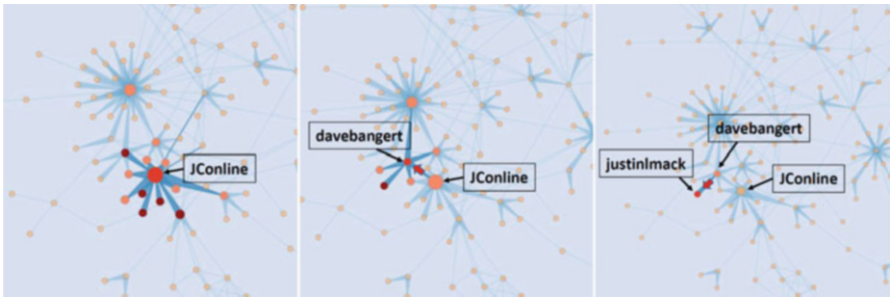


Fig. 7 Iterative approach to locate influential users. Five users serve as the source of information for *JConline* (*left*). The messages retweeted by *JConline* are propagated to other users (*center* and *right*)

which contain pictures of the event. Furthermore, most of the messages are later retweeted by *JConline* and then quickly spread to other users. Accordingly, law enforcement analysts can maintain a situational awareness of an evolving situation, and locate and track the activity of different user groups from the social networks in order to rapidly collect important information pertaining to the event.

6 Conclusion

In social media analysis, identifying influential users and analyzing user interaction and information diffusion in the networks can improve situational awareness of events and provide significant and reliable information in emergency management. Analysts working on such issues often need to look into the contextual evidence that could help them create hypothesis. To provide such contextual information, we introduced our visualizations of user influence-based dynamic social networks, that make it possible to identify user-influence from social networks and analyze information diffusion in social networks. Also, we described our visual analytics system to cope with large amounts of messages and help analysts to explore and detect important messages. Our system combines visual presentation with data mining and statistical models in order to take advantage of the synergistic impact of the multiple techniques. The system allows users to directly manipulate the algorithms, easily understand results of the algorithms and how the algorithms work in order to facilitate knowledge discovery from social media data by enabling analysts to generate, test, and refine hypotheses from data. We integrate two techniques including the LDA topic model and the time series decomposition statistical technique with our analysis environment to detect abnormal events. We demonstrated the usage and effectiveness of our system for social network analysis and anomaly analysis in abnormal situations by case studies.

References

Andrienko, G., Andrienko, N., Bosch, H., Ertl, T., Fuchs, G., Jankowski, P., et al. (2013). Thematic patterns in georeferenced tweets through space-time visual analytics. *Computing in Science and Engineering, 15*(3), 72–82.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research, 3*, 993–1022.

Bosch, H., Thom, D., Worner, M., Koch, S., Puttmann, E., Jackle, D., et al. (2011). Scatterblogs: Geo-spatial document analysis. In *IEEE Symposium on Visual Analytics Science and Technology* (pp. 309–310).

Card, S. K., & Nation, D. (2002). Degree-of-interest trees: A component of an attention-reactive user interface. In *Proceedings of the Working Conference on Advanced Visual Interfaces, AVI'02* (pp. 231–245). New York: ACM. doi:10.1145/1556262.1556300.

Chae, J., Thom, D., Bosch, H., Jang, Y., Maciejewski, R., Ebert, D., et al. (2012). Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In *IEEE Symposium on Visual Analytics Science and Technology, October 14–19, 2012* (pp. 143–152).

Chae, J., Thom, D., Jang, Y., Kim, S., Ertl, T., & Ebert, D. S. (2014). Public behavior response analysis in disaster events utilizing visual analytics of microblog data. *Computers & Graphics, 38*, 51–60.

Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association, 74*(368), 829–836.

Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. (1990). STL: A seasonal-trend decomposition procedure based on loess (with discussion). *Journal of Official Statistics, 6*(1), 3–73.

Correa, C., Crnovrsanin, T., & Ma, K.-L. (2012). Visual reasoning about social networks using centrality sensitivity. *IEEE Transactions on Visualization and Computer Graphics, 18*(1), 106–120.

Crnovrsanin, T., Liao, I., Wu, Y., & Ma, K.-L. (2011). Visual recommendations for network navigation. *Computer Graphics Forum, 30*, 1081–1090.

Dork, M., Carpendale, S., Collins, C., & Williamson, C. (2008). VisGets: Coordinated visualizations for web-based information exploration and discovery. *IEEE Transactions on Visualization and Computer Graphics, 14*(6), 1205–1212.

Fayyad, U. M., Piatetsky-Shapiro, G., & Smith, P. (1996). From data mining to knowledge discovery: An overview. In U. M. Fayyad, et al. (Eds.), *Advances in knowledge discovery and data mining* (chap. 1, pp. 1–34). Menlo Park, CA: AAAI Press/MIT Press.

Field, K., & O'Brien, J. (2010). Cartoblography: Experiments in using and organising the spatial context of micro-blogging. *Transactions in GIS, 14*, 5–23. doi:10.1111/j.1467-9671.2010.01210.x.

Fruchterman, T. M. J., & Reingold, E. M. (1991). Graph drawing by force-directed placement. *Software Practice and Experience, 21*(11), 1129–1164. doi:10.1002/spe.4380211102.

Ho, C.-T., Li, C.-T., & Lin, S.-D. (2011). Modeling and visualizing information propagation in a micro-blogging platform. In *IEEE International Conference on Advances in Social Networks Analysis and Mining (ASONAM)* (pp. 328–335).

Keim, D. A. (2002). Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics, 8*(1), 1–8.

Krovetz R. (1993). Viewing morphology as an inference process. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'93* (pp. 191–202). New York: ACM. doi:10.1145/160688.160718.

Krueger, R., Thom, D., & Ertl, T. (2014). Visual analysis of movement behavior using web data for context enrichment. In *Proceedings of the 2014 I.E. Pacific Visualization Symposium, PACIFICVIS'14* (pp. 193–200). Washington, DC: IEEE Computer Society.

MacEachren, A., Jaiswal, A., Robinson, A., Pezanowski, S., Savelyev, A., Mitra, P., et al. (2011). SensePlace2: GeoTwitter analytics support for situational awareness. In *IEEE Symposium on Visual Analytics Science and Technology* (pp. 181–190).

Macskassy, S. A., & Michelson, M. (2011). Why do people retweet? Anti-homophily wins the day!. In *International AAAI Conference on Web and Social Media*.

McCallum, A. K. (2002). *Mallet: A machine learning for language toolkit*. Retrieved from http://mallet.cs.umass.edu

Norman, D. A. (1993). *Things that make us smart: Defending human attributes in the age of the machine*. New York: Basic Books.

Romero, D. M., Galuba, W., Asur, S., & Huberman, B. A. (2011). Influence and passivity in social media. In *Machine learning and knowledge discovery in databases* (pp. 18–33). Berlin, Germany: Springer.

Roth, E., & White, J. (2010) Twitterhitter: Geovisual analytics for harvesting insight from volunteered geographic information. In *Proceedings of GIScience*.

Sakaki, T., Okazaki, M., & Matsuo, Y. (2010). Earthquake shakes twitter users: Realtime event detection by social sensors. In *Proceedings of the 19th International Conference on World Wide Web, WWW'10* (pp. 851–860). New York: ACM.

Shneiderman, B. (1996). The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the IEEE Symposium on Visual Languages* (pp. 336–343).

Shneiderman, B. (2002). Inventing discovery tools: Combining information visualization with data mining. *Information Visualization, 1*(1), 5–12.

Suh, B., Hong, L., Pirolli, P., & Chi, E. H. (2010). Want to be retweeted? Large scale analytics on factors impacting retweet in Twitter network. In *Proceedings of the 2010 I.E. Second International Conference on Social Computing (SocialCom)* (pp. 177–184).

Sun, J., Papadimitriou, S., Lin, C.-Y., Cao, N., Liu, S., & Qian, W. (2009) Multivis: Content-based social network exploration through multi-way visual analysis. In *SDM* (Vol. 9, pp. 1063–1074). SIAM.

Terpstra, T., Stronkman, R., de Vries, A., & Paradies, G. (2012). Towards a realtime twitter analysis during crises for operational crisis management. In *Proceedings of the 9th International ISCRAM Conference, Vancouver*.

Thom, D., Bosch, H., Koch, S., Woerner, M., & Ertl, T. (2012). Spatiotemporal anomaly detection through visual analysis of geolocated twitter messages. In *IEEE Pacific Visualization Symposium (PacificVis)* (pp. 41–48).

Wakamiya, S., Lee, R., & Sumiya, K. (2011). Crowd-based urban characterization: Extracting crowd behavioral patterns in urban areas from twitter. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks, LBSN'11*. New York: ACM.

Yang, X., Asur, S., Parthasarathy, S., & Mehta, S. (2008). A visual-analytic toolkit for dynamic interaction graphs. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1016–1024). New York: ACM.

Ying, J. J.-C., Lee, W.-C., Ye, M., Chen, C.-Y., & Tseng, V. S. (2011). User association analysis of locales on location based social networks. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks, LBSN'11* (pp. 69–76). New York: ACM.

A Survey of Visual Analysis Approaches for Financial Data Exploration

Anonymous

Abstract—Market participants and businesses have made tremendous efforts to make best the decisions on time in varying economic and business circumstances. Therefore, the decision-making processes based on financial data have been a popular topic in industries. But, analyzing financial data is a non-trivial task due to large volume, diversity and complexity, and this has led to rapid research and development of visualizations and visual analytics systems for financial data exploration. Often, the development of such systems requires researchers to collaborate with financial domain experts to better extract requirements and challenges in their tasks. However, there has not been work to systematically study and gather the task requirements and to acquire an overview of existing visualizations and visual analytics systems that have been applied in financial domains with respect to real-world data sets. To this end, we perform a comprehensive survey of visualizations and visual analytics. In this work, we categorize financial systems in terms of data sources, applied automated techniques, visualization techniques, interaction, and evaluation methods. In addition, we present task requirements extracted from interviews with domain experts in order to help researchers design with detailed goals.

Index Terms—State-of-the-art, finance visualization, visual analytics, survey, business visualization

1 Introduction

The availability of different sources of financial data provides opportunities to market analysts and investors to extract new insights in order to make informed decisions. Such analyses guide them in the development of optimal investment and risk management strategies. To facilitate such analysis tasks, the analysts traditionally utilize visual analysis tools that are typically built atop standard statistical methods (e.g., moving averages and regression). Examples of these tools include line graphs and candlestick charts [1] that are popular among financial market professionals for decision making tasks [2].

Despite the popularity of these techniques, these tools are often inadequate to handle the problems that arise from big data. With the advent of the digital age, an unprecedented amount of real-time financial data is generated in different sectors, including asset trading, news, and economic indicators. Thousands of stocks with multivariate attributes (e.g., price, time, volume) get traded every few seconds, and various reports on the stock market and economic indicators are continuously published for stakeholders. In the big data era, the conventional approaches of analyzing these datasets using standard analysis techniques are often marred with limitations (e.g., scalability, overplotting issues). Additionally, analysts are often requires to explore and consider discrete and fragmented pieces of information at the same time for making informed decisions.

The complexity and size of these financial datasets have attracted the attention of the information visualization and visual analytics communities. This has led to the rapid development of many visual analytic systems and interaction methods for (e.g., bank wire transaction [3], corporate sales [4]). The development of such systems often requires visual analytics researchers to collaborate with financial experts to assess their requirements and challenges and design appropriate visual analytics solutions to address them. This provides the impetus to systematically study and gather the needs and requirements of financial practitioners from different domains and to gain an overview of the existing visual analytics techniques that have been applied with respect to their datasets.

To address this need, researchers have conducted several surveys that provide an overview of the existing works in the financial domain. For example, the survey conducted by Pryke [5] discusses the role of visualization techniques in financial industry applications. Schwabish's survey [6] provides an introduction and guidance to economists. More recently, Flood et al. [7] survey applications designed for monitoring financial stability. However, these surveys are limited in scope in terms of the number of systems surveyed and the financial domain targeted by the researchers. To this end, we aim to perform a comprehensive survey of research papers in the information visualization and visual analytics domains that target the financial market sector. In this work, we categorize financial systems in terms of data sources, applied automated techniques and visualization techniques, interaction methods, and evaluation. We organize the paper by examining different facets of the tasks involved, including the user requirements, data sources, applied automated techniques, visualization techniques, interaction methods, and evaluation methods.

The contributions of this work are two-fold. First, we identify and summarize general financial analysis task requirements by interviewing analysts with a background in financial analysis, risk management, and capital market investment management. Second, we perform a comprehensive survey of recent developments in the visual analytics domain and categorize previous approaches based on the input data, the automated techniques, visualizations, interaction, and evaluation methods. We concentrate mainly on research publications and also include visual analysis tools from the financial industry if enough information can be acquired from the research publications. Our work is intended to assist financial analysts and researchers in their common tasks, which include the

- *Anonymous*
  E-mail:
- *Anonymous are with Anonymous University.*

Manuscript received April 19, 2005; revised September 17, 2014.

TABLE 1
16 typical business application domains [8].

Financial Risk Management	Industrial Process Control
Operations Planning	Capital Markets Management
Military Strategic Planning	Network Monitoring
Marketing Analysis	Derivatives Trading
Fraud/Surveillance Analysis	Portfolio Management
Actuarial Modeling	Customer/Product Analysis
Budget Planning	Operations Management
Economic Analysis	Fleet/Shipping Admin

following: 1) identify financial data sources used by stakeholders in their analysis and decision making, 2) provide an overview of data-mining and visualization techniques applied to produce desired results, 3) understand common user interaction approaches for analyzing financial data, and 4) evaluation strategies that are applied to visual analytic systems to show usability of the systems.

This paper is structured as follows. First, we present related work in the next section. Then, we categorize previous work based on data type (Section 5), applied data mining (Section 6), visualization (Section 7), and interaction techniques (Section 8). In Section 9, we compare how the previous works have been evaluated and discuss trends, findings and insights acquired from surveying papers in Section 10. Finally, we conclude our work by summarizing our findings and possible future work.

2 BACKGROUND

In this section, we discuss related surveys for visual financial data analysis. A survey conducted by Tegarden [8] is the first survey where information visualization techniques in industry systems are highlighted. This work is important in that it defines 16 typical business application domains as shown in Table 1. There are two surveys from perspectives of economics [5], [6]. With a goal to examine implications of visualizations on industry applications, Pryke [5] conducted interviews with representatives of software companies and financial organizations and presents financial industry applications. After examination of applications based on interviews, Pryke concludes that visual approaches should be more highlighted for financial market analysis. However, it is unfortunate that little information is revealed about task requirements from the interviews due to a proprietary issue as described in the appendix. The work conducted by Flood et al. [7] is different from others in that Flood's work mainly focuses on systems for monitoring financial stability and benefits of Visual Analytics [9] in performing the monitoring tasks. Dumas et al. [10], [11] provide an excellent on-line visual overview with filtering options for financial visualization systems.

Compared to the surveys described above, there are three benefits in our work. We provide the most comprehensive and detailed survey on visualizations techniques and visual analytics approaches that have been applied to financial data. We classify previous systems for financial data analysis into several categories: data sources, automated techniques, visualizations, interactions, and evaluations and derive patterns and insights for each category. In addition, we present task requirements based on interviews with a number of analysts whose backgrounds are varied. We think that the presented requirements are able to complement low-level analysis operations in tasks (9 operations in [12] and 6 operations in [13]). We believe that our work is able to support industry experts in finding the best suitable visualization techniques and visual analytics approaches for their data and to help researchers in the visualization and analytics communities understand what functionalities or visualizations are needed in the tasks for various business domains.

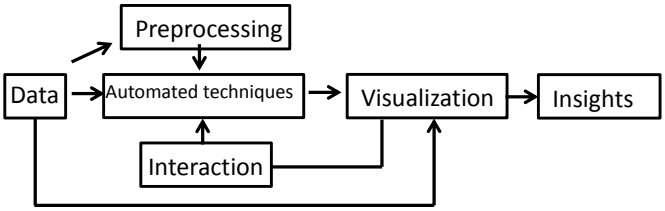


Fig. 1. Financial data exploration pipeline. In financial visual analytics systems, a loop is formed between automated techniques and visualizations through user interaction, presenting the "human-in-loop" aspect of analysis [14]. In the following sections, visualizations and analysis techniques are categorized for each block except the preprocessing stage.

3 METHODOLOGY

This section provides an overview of visualizations and visual analytics systems for financial data. While we were surveying papers, we found that there is little information available about task requirements that are derived directly from financial analysts. We found one paper presenting types of action in tasks for a user studies [12]. We believe such sparse information can limit researchers from acquiring detailed task requirements for their research. In order to mitigate this problem, we interviewed analysts from different financial sectors in order to systematically gather the general domain requirements for visual analytics researchers. These task requirements are presented in Section 4. The discussions, interviews, and task elicitation processes contribute to deriving a visual analysis pipeline.

Visual data exploration models [9], [14] and the information foraging loop [15] have been fundamental concepts in designing visual analytics systems. In order to facilitate our discussion, we adapt the visual analysis pipeline model proposed by Keim et al. [14] for financial datasets. This is shown in Figure 1. During our reviews and discussions with the analysts, we identified and categorized the various tasks using this model. In later sections, we will categorize previous visualizations and visual analytic systems using the scheme presented in Figure 1.

During reviews, we found that an additional preprocessing step was often required for certain data sets such as pattern detection or topic modeling (e.g., time-series stock data linked to news stories). Then, data are ingested directly into automated technique modules that are then followed by data visualizations. The flow between automated techniques and visualizations is realized by various interaction methods that enable the "human-in-the-loop" in the visual analytics process.

For the paper collection, we started from previous surveys described in Section 2 with a focus on papers that incorporate visualization techniques in the analysis loop. Then, we utilized search tools, as well as the IEEE Xplore and ACM digital libraries with keywords combinations of "visualization", "financial", "economy", "exploration", and "analysis". We also individually explored prestigious venues for publications as shown in Table 2 and Figure 3 for related work in this domain. We selected papers that had a focus on performing a visual analysis of financial data. In addition, we chose newer versions of papers if similar approaches were used from the same research groups. In the end, we included 50 papers in total for our review. We provide a summary

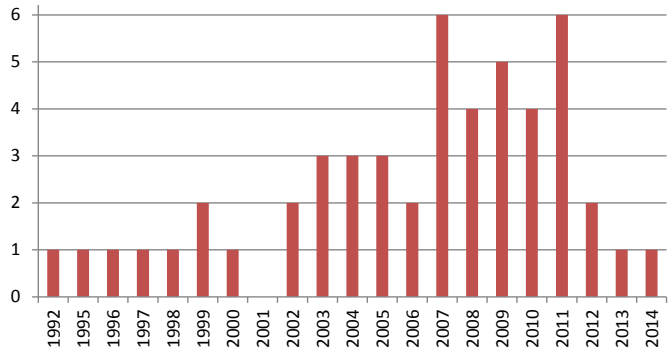


Fig. 2. Distribution of papers by publication years. Many papers were published between 2007 and 2011.

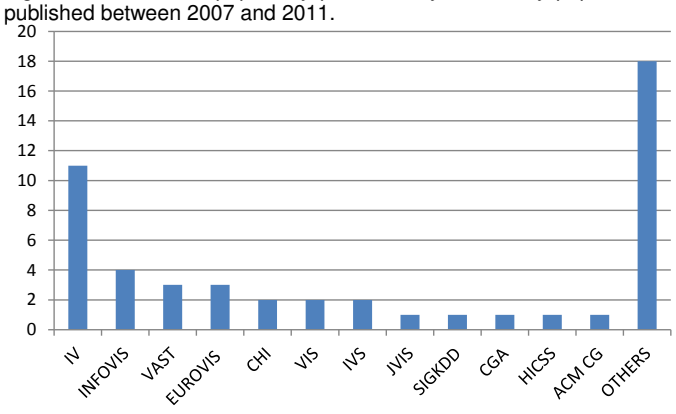


Fig. 3. Number of papers published from venues. The abbreviations of the venues are explained in Table 2. "Others" sums all venues other than those in Table 2.

distribution of papers over their corresponding publication years in Figure 2. We also present the results of our survey on financial data analysis systems in the following tables. We group papers by year and sort them alphabetically by first author name, placing each paper in the appropriate category. If at least one industry expert coauthored a paper (judged by affiliation), we color the paper author column dark blue to indicate collaborations between researchers and industry experts. We also provide the sum of the papers that have been published in collaboration with industry experts in the "From industry" column.

4 Survey of Task and System Requirements for Financial Data Analysis

In this section, we present a summary of system requirements from our discussions with financial analysts and from our survey. We interviewed domain experts from the financial risk management, economic analysis, and capital market management areas to obtain task requirements. Even though the requirements summarized in this section have mainly been acquired from experts from a subset of the financial industry, we believe that they are representative of other financial domains as well.

We held two work flow and process design sessions with analysts (task-level knowledge elicitation) and three focus group sessions to obtain general system requirements, with the focus group size ranging from 2 to 9 analysts. The analysts had diverse backgrounds from economics and social science to quantitative analytics and statistics. Their current roles also varied including many who were interested in predictive analytics.

The first focus group is responsible for detecting anomalies from financial datasets (e.g., unusual variations in stock prices)

TABLE 2
Searched venues for paper collection.

IV	International Conference on Information Visualization
InfoVis	IEEE Symposium on Information Visualization
VAST	IEEE Conference on Visual Analytics Science and Technology
EuroVis	Joint Eurographics-IEEE Symposium on Visualization
CHI	ACM Conference on Human Factors
VIS	IEEE Conference on Visualization
IVS	Information Visualization (published by SAGE)
JVIS	Journal of Visualization
SIGKDD	ACM Conference on Knowledge Discovery and Data Mining
CGA	Computer Graphics and Applications
HICSS	Hawaii International Conference on System Sciences
CG	ACM Conference on Computer Graphics

and generating reports with evidence for further action. Analyzing such anomalies is itself a complex and time-consuming task. It requires analysts to look into events that have contextual evidence available through news data sources. It also requires analysts to understand historical patterns and causes of events and to distinguish whether events have regular or periodic properties before the analysts can decide whether the events are considered to be anomalous. There is also a need to show statistical significance of the detected anomaly for guiding decision-making tasks. The second focus group consisted of analysts responsible for generating reports on future financial trends. While the initial procedure of detecting anomalies is similar to that of the first, this group focuses on making a large number of comparisons among anomalies in the market data attributes to infer unknown facts.

Next, we interviewed three analysts from a large U.S. bank with backgrounds in computer science, economics, and mathematics who are tasked with finding patterns and trends from financial data and reporting results to decision makers. Their main duties include estimation of retail sales using customer transaction data combined with U.S. Census data. The analysts expressed that due to the size and complexity of the recent datasets, various machine learning techniques are used including k-means, principle component analysis, support vector machine, and linear and logistic regression to facilitate the analysis. In addition, several visual tools (e.g., Tableau and Spotfire) and statistical packages (e.g., SAS and R) are extensively used due to their effectiveness in exploring data. When asked which type of action is hard, they expressed that the majority of the difficulty during their tasks comes the need to spend a lot of time to shape data in a digestible size, clean data, and customize the their analysis results for presentation. A manager who leads a team of 6 analysts has additional tasks such as verifying data security, ensuring customer privacy, and removing unnecessary information encumbering analytics processes.

Finally, we interviewed an economist responsible for analyzing large scale financial trends (e.g., sector, nation-wide trends). This analyst's requirements were similar to those of the previous groups; however, the analyst stressed the importance of providing contextual evidence about anomalies that helps confirm certain hypotheses. This analyst was also interested in comparing the anomalies across different aggregation levelsand in support for users seeking explanations regarding anomalies. Based on these discussions, we summarize detailed requirements for visual data analytics systems for financial data exploration. In general, a visual analytics system for financial data should:

R1 Provide ample information including historical and contextual data in order to help users find historical and recurring patterns, and causes of events,

	1992	1995	1996	1997	1998	1999	2000	2002		2003				2004		2005			2006	2007						2008			2009				2010				2011						2012	2013	2014	Sum	From Industry
	Jungmeister92 [19]	Strausfeld95 [20]	Ankerst96 [21]	Brodbeck97 [22]	Kirkland98 [23]	Gresh99 [24]	Wattenberg99 [25]	Groenen00 [26]	Eklund02 [27]	Xiong02 [28]	Csallner03 [29]	Simunic03 [30]	Taskaya03 [31]	Dwyer04 [32]	Nesbitt04 [33]	Shen04 [34]	Hao05 [35]	Lin05 [36]	Smeulders05 [37]	Keim06 [38]	Schreck06 [39]	Chang07 [3]	Mirel07 [40]	Schreck07 [41]	Tekusova07 [42]	Ziegler07 [43]	Ziegler07a [44]	Chiu08 [45]	Dao08 [46]	Tekusova08a [47]	Ziegler08 [48]	Alsakran09 [49]	Huang09 [50]	Rudolph09 [51]	Sawant09 [52]	Schreck09 [53]	Alsakran10 [54]	Lei10 [55]	Wu10 [56]	Ziegler10 [57]	Gross11 [58]	Lei11 [59]	Sarlin11a [60]	Sarlin11b [61]	Savikhin11 [62]	Schaefer11 [63]	Brath12 [64]	Ko12 [4]	Sorenson13 [2]	Lemieux14 [65]		
Stock	■		■				■	■				■	■	■		■	■			■				■	■				■				■		■	■						■			■	■		■		23	6
Fund		■							■	■	■									■						■	■				■	■					■													9	3
Econ. Indi.																						■							■												■		■							4	2
Trans.				■	■	■									■			■				■																						■						7	5
Risk																																	■															■		3	0
Company Info								■									■		■									■																				■		5	1

TABLE 3
Categories for data sources. We derive 6 data sources. It is obvious that stock and fund data are the most popular data in both research communities and financial industries. Some data were underutilized due to possible confidentiality or privacy in data (e.g., customer data for marketing).

R2 Support users with automated techniques for detecting anomalous patterns or situations,
R3 Allow interaction methods for drill-down analysis,
R4 Present whether an anomaly (if detected automatically) is statistically significant,
R5 Forecast future trends based on user selected data ranges,
R6 Provide functions for data cleaning, data shaping, and result customization for presentation.

Here, by the contextual data in R1, we mean other auxiliary data that are oftentimes considered along with the main target data. For example, the analysts stated in the interviews that they often combine news stories in analyzing stock data to gather contextual evidence to support or reject their hypotheses. In R3, various interaction methods exist based on different operations [16] and tasks [17], and the analysts are especially interested in adjusting parameters of the applied data mining techniques (e.g., thresholds for anomaly judgment) and verifying the results in visualizations on the fly.

It should be noted that Kohavi et al. [18] also discuss requirements in terms of time and performance constraints in business analytics systems (e.g., time for analysis cycle, integration of data, distribution of results). However, compared to Kohavi's work, the requirements discussed in this section provide task requirements and functionalities of systems at both an overview and detail level. In the next sections, we present our review on different aspects including data type, visualization techniques, interaction methods, and evaluation.

5 Financial Data Sources

In this section, we summarize and categorize data sources used in financial analysis systems as follows: stocks, funds, economic indicators, transactions, risk, and company information as shown in Table 3. Note that while stock and fund data are the data relevant to stocks, the others have a relatively broad coverage. For example, the company information category covers all the information related to a business (e.g., profit, financial statements, sales, and marketing data) and the transaction category considers all the data generated by transactions among different subjects (e.g., bank customers, companies, and countries). In a perspective from the business domains (Table 1), visualizations for these data sources can cover the domains of financial risk management, economic analysis, capital markets management, portfolio management, and marketing analysis. Stock data containing time series of share prices of companies over a long time have been extensively used in financial data analyses including trend, pattern, performance, and predictive analysis. It is notable that the number of stocks used in analyses ranges from few to hundreds [19], [20], [21], [22], [23], [24], and to several thousand stocks (5328 stocks in a visualization) [25]. The stock data is often combined with news media data [26], [27] to extract additional knowledge for providing contextual information.

Analyzing groups of stocks is also popular (i.e, sector analysis and fund analysis). In sector analysis, stocks are aggregated based on industry-defined sectors (e.g., energy, healthcare, information technology) to be used in the analysis, while fund analysis uses groups of stocks gathered by investment consideration (e.g., mutual funds). Note that the data of groups of stocks compiled for industry sectors or funds possess hierarchies that are base data for hierarchy visualization techniques such as treemaps [28].

Economic indicators are statistical information and affect financial markets with other factors (e.g., political news and corporate earning news). A basic approach in economic indicator analysis is to rely on a single indicator in the market such as consumer price indexes [29] or inflation [30]. Compared to this approach, Sarlin [31] uses 35 general and economic variables for monitoring debt ratios, which is the largest number of variables used for indicator visualization. Transactions are generated in many financial domains including bank, investment, and money exchange industries. Transaction analysis is important because it could reveal unknown transactional patterns, malfunctions in a business, and evidence or symptoms of security breaches. Many types of the transaction data have been analyzed with visualizations such as the transactions among customers (e.g., wire transfer [3], fixed income trading [32]), businesses (e.g., bids and asks [33], [34] or option trading [35]), and countries (e.g., currency exchange rates among 23 countries [36], [37]).

Risk analysis is one important parts of business and investment, but the analysis of risk with visualizations is most underutilized. Among three research projects, two projects [38], [39] show risk data generated from an economic model in user studies in order to find whether investors are able to recognize the risk in their investment and change their investment strategies according to the risk. In the third project, Lemieux et al. [40] design a visual analytics system that visualizes multiple types of data including credit default swaps to help managers in banks understand systemic risk in the domain.

Often, accessibility to data affects the focus of research. More

	1997	1998	2000	2003	2004	2005	2007	2008	2009		2010	2010	2011			2012		
	Brodbeck97 [22]	Kirkland98 [23]	Groenen00 [26]	Simunic03 [30]	Dwyer04 [32]	Lin05[36]	Schreck07 [41]	Ziegler08 [48]	Rudolph09 [51]	Schreck09 [53]	Lei10 [55]	Ziegler10 [57]	Savikhin11[62]	Sarlin11a [60]	Sarlin11b [61]	Ko12 [4]	Sum	From Industry
K-means											■	■			■		3	1
SOM				■			■			■				■	■		5	3
MDS	■		■														2	1
PCA					■												1	0
Sampling						■											1	0
Moving Avg.																■	1	0
Regression																■	1	0
Decision Tree		■															1	1
Specific Models								■	■				■				3	1

TABLE 4
Categories for automated techniques. We categorize automated techniques used for financial data into 4 categories–clustering, dimensional reduction, statistical, and model- or theory-based techniques. For financial analysis, SOM (self-organizing maps) was used for both clustering and dimension-projection.

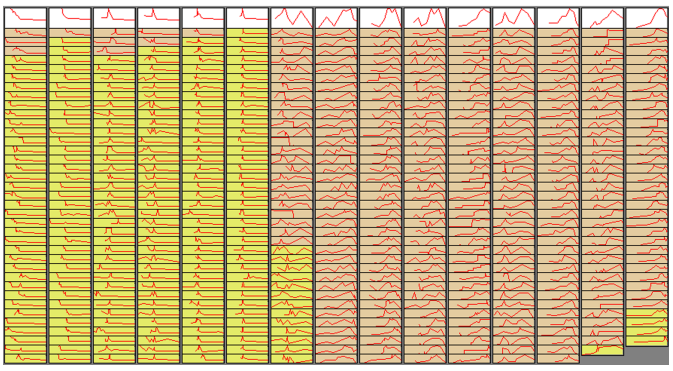


Fig. 4. Trajectory-based clusters are compared for analysis. Original data include 550 assets from sectors of software (yellow) and banks (brown) from 05/1994 to 06/2010 [43].

than half of the papers use stock and fund data. This is expected due to easy access to financial data through on-line financial web sites. Therefore, stock data have been most utilized from early stages of financial visualizations. Compared to stock data, not much work has been done for visualizing economic indicators, although they are also accessible on-line.

It is notable that visual analytics systems for analyzing transaction data were designed only when an author from an industry participated in the project. This is expected, because such data tend to be confidential and would have limited access. However, we rarely find work performed with corporations for analyzing risks and company information. This came as a surprise initially because it is known that performance and risk evaluation of companies are very important areas for competitive intelligence [41]. Additionally, marketing research provides insights on customers' fast-changing trends, demand and buying patterns. We found only one paper [42] utilizing visualizations of customer data for marketing research. On the other hand, this can be understood since corporations tend not to expose projects they are interested in due to competition in the market and privacy issues existing in companies' data.

6 Applied Automated Techniques

There have been a number of automated techniques utilized for visual analysis of financial data as shown in Table 4. We recognize that these techniques are applied to financial data with purposes of clustering, dimensional reduction, trend and pattern analysis, and forecast.

Clustering is a process of dividing data into groups of similar objects [45]. Even though it may lose detailed information, clustering effectively reduces a size of data for visualizations. Efficiency of a clustering algorithm depends on applications. This leads to development of a variety of methods on different criteria. For financial data visualizations, k-means [46] and Self-Organizing Map (SOM) [47] are mainly utilized. In the k-means clustering, each of the clusters is presented by a mean (or weighted average) of data points in each cluster, the so-called centroid. Then, the discrepancies between the data points and the centroid are used to form clusters. With numerical data, this approach has an advantage in providing a good statistical and geometry sense in the results. For example, Ziegler et al. [43] utilize the k-mean algorithm for generating clusters with 550 assets due to its high speed computation, easy implementation, and ability to specify a desired amount of clusters as shown in Figure 4. In order to produce clusters for analyzing stock data based on trading patterns, Lei and Zhang [48] utilize k-core, a variant of k-means, that allows pruning nodes and links whose degrees are less than k.

SOM [47] is a neural network based on unsupervised learning techniques and that gained popularity for a unique reason. As a clustering algorithm (e.g., k-means clustering), the SOM produces topological clusters; similar clusters are neighboring. On the other hand, SOM projects multidimensional data into a two-dimensional grid as a projection technique. Therefore, SOM generates both clusters and topology-preserved mapping of prototype vectors on a grid structure in low dimensional space. Many visualization systems incorporating SOM tend to use the generated mapping data when the clusters are placed. For example, the approach shown in [19], [21], [21] presents how SOM can be used for a clustering purpose with stock data while the work while the approach in [37] demonstrates how the mapping information can be utilized for placing clusters generated from multivariate economic indicators.

Sometimes, the analysis with multivariate indicators can be easier if the multidimensional data can be placed in a lower dimensional space. To this end, two dimensional reduction techniques are used for financial data exploration: MultiDimensional Scaling (MDS) [49] and Principle Component Analysis (PCA) [50]. Even though the notion of projecting multidimensional data onto a lower dimensional space is similar, there are differences between these two algorithms. MDS aims at presenting a proximity matrix based on Euclidean distances for a 2D or 3D space. On the other hand, PCA seeks to find the axes of greatest variance (i.e., principal component) and to capture the variance by projecting the data onto plane. In addition, PCA could be more practical when scalability is an issue with large data [51]. When these projection techniques are used for transaction [32] and stock markets data [20], [52], scatterplot-based visualizations are often utilized because the results of the algorithms need to be projected onto a 2D [32], [52] or 3D [20] grid.

There are techniques that are rarely used in financial research work including sampling, moving average, regression, decision

	1992	1995	1996	1997	1998	1999		2000	2002		2003			2004			2005		2006	2007						2008			2009				2010				2011						2012		2013	2014	Sum	From Industry
	Jungmeister92 [19]	Strausfeld95 [20]	Ankerst96 [21]	Brodbeck97 [22]	Kirkland98 [23]	Gresh99 [24]	Wattenberg99 [25]	Groenen00 [26]	Eklund02 [27]	Xiong02 [28]	Csallner03 [29]	Simunic03 [30]	Taskaya03 [31]	Dwyer04 [32]	Nesbitt04 [33]	Shen04 [34]	Hao05 [35]	Lin05 [36]	Smeulders05 [37]	Keim06 [38]	Schreck06 [39]	Chang07 [3]	Mirel07 [40]	Schreck07 [41]	Tekusova07 [42]	Ziegler07 [43]	Ziegler07a [44]	Chiu08 [45]	Dao08 [46]	Tekusova08a [47]	Ziegler08 [48]	Alsakran09 [49]	Huang09 [50]	Rudolph09 [51]	Sawant09 [52]	Schreck09 [53]	Alsakran10 [54]	Lei10 [55]	Wu10 [56]	Ziegler10 [57]	Gross11 [58]	Lei11 [59]	Sarlin11a [60]	Sarlin11b [61]	Savikhin11 [62]	Schaefer11 [63]	Brath12 [64]	Ko12 [4]	Sorenson13 [2]	Lemieux14 [65]		
Std (2D)																																																11	2
Std (3D)																																																11	4
GT																																																10	6
Iconic																																																6	2
Dense Pixel																																																13	3
Stacked																																																7	3

TABLE 5
Categories for visualization techniques. We categorize the papers according to the taxonomy in [44]. As we expected standard charts have been a popular technique in analysis. Among advanced visualizations, dense and geometrically-transformed displays were used more than iconic and stacked.

tree, and risk/return models borrowed from economics. Sampling is a statistical process where a predetermined number of samples are taken from a large data. How many samples are taken from data depends on the type of analysis. The work by Lin et al. [36] is the only work using a sampling technique. In their work, the sampling technique is utilized in order to reduce data size in a stream of financial data. Applying the moving averages technique basically helps reduce the impact of noise in the data. More importantly for financial analysis, it also can be used for trend analysis combined with regression techniques. Therefore, it is not a surprise that the system proposed by Ko et al. [4] incorporates these two techniques for predicting future sales data for analyzing competitive advantages of corporates.

A decision tree is a technique utilizing a tree-like shape where each branch in the tree represents a consequence of a decision or event. This technique is useful when a complex decision-making process can be proceeded with a series of selections among possible answers based on probability. The ADS (Advanced Detection System) [33] is a good example that shows the effectiveness of the decision-tree. The system has been used since 1997 to monitor regulatory breaks in stock trading, and it has detected over 7000 breaks. There are two research projects that utilize models originated from economics [38], [39] in order to visualize results of an investor's decisions. Both systems are used for user studies to find if investors can make better decisions with the support of a visual analytics system that visualizes the results of their investment as feedback.

We found two patterns in applying automated techniques. Often, some techniques have been used for making data more understandable by directly reducing a data size (e.g., sampling) or grouping data into subsets of data (i.e., clustering and dimensional reduction). In many use case scenarios, it is observed that users proceed their exploration with interaction methods (e.g., abstract and filter in Section 8) after these techniques are applied. We also confirm that the number of utilized automated techniques is low when we consider other available dimensional reduction [53] and clustering [45], [54] algorithms in the data-mining research area. On the other hand, some trend analysis techniques (e.g., moving averages and regression) are not extensively used in financial visualization researches. This is a surprise to us because domain experts whom we have interviewed mentioned that they have extensively used automated statistical techniques in their work as described in Section 4. They all have a notion in mind that accurate prediction of values and trends, and the verification of changes in data or human activity could be very helpful in designing future strategies and producing high returns with low risks in investments. Therefore, we think that when many businesses use a variety of techniques for a type of task (e.g., finding an anomaly), they try to avoid revealing their techniques in publications.

7 Visualization Techniques

Our survey reveals that financial data mainly tend to be a set of time-series data with attributes as discussed in Section 5. In addition, we find that various visualization techniques have been applied to financial data, including parallel coordinate plots [55], treemaps [56], scatterplot matrices [57], glyph based visualization techniques [58], stacked displays [59], and dense displays [60], [61]. In this section, we utilize Keim's [44] classification to categorize the different approaches utilized for financial data: 2D/3D displays, geometrically transformed displays, iconic displays, dense pixel displays, and stacked displays. When multiple visualization techniques are used in a paper, we categorize the paper based on the primary visualization technique for analysis. For each category, we add sub-categories if necessary in order to present differences within the same-category techniques.

7.1 Standard visualization techniques

The most common technique for visualizing financial data remains in the family of standard charts (e.g., line, bar, and pie charts and their variations [62]). These standard charts utilize 2D data whose basic idea is to present attribute values along the y-axis and time on x-axis. 3D visualization techniques can facilitate the analysis of data by adding an essential dimension to another axis. Although these charts help users in finding trends and key events (e.g., spikes), they suffer from issues of over-plotting, occlusion, and insufficient support for comparing multiple datasets [2]. In addition, these are not appropriate for analyzing higher dimensional data. In subsequent sections, we discuss these visualization techniques.

7.1.1 2D Display

2D charts have been extensively used in the analysis of financial datasets. The basic use of these charts is an auxiliary visualization in multiple coordinated views (e.g., line graph view for stock prices) [3], [27], [63]. The line graphs are also used in financial

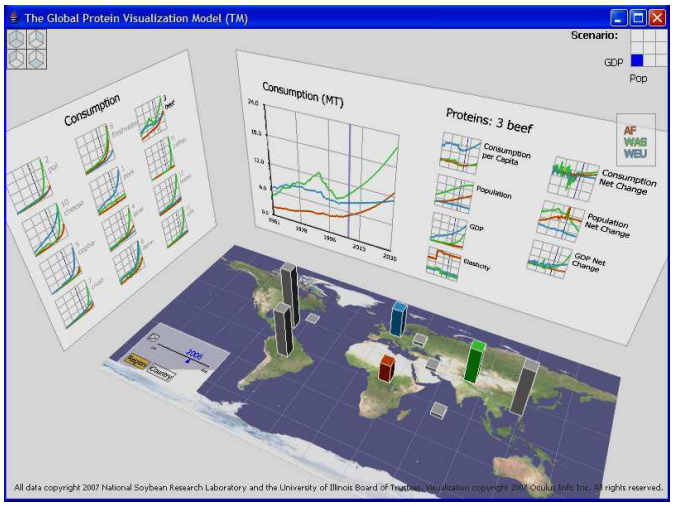


Fig. 5. An example of analysis of commodity market in 3D. This workspace combines 2D line charts and 3D bar (or column) charts in a workspace [66].

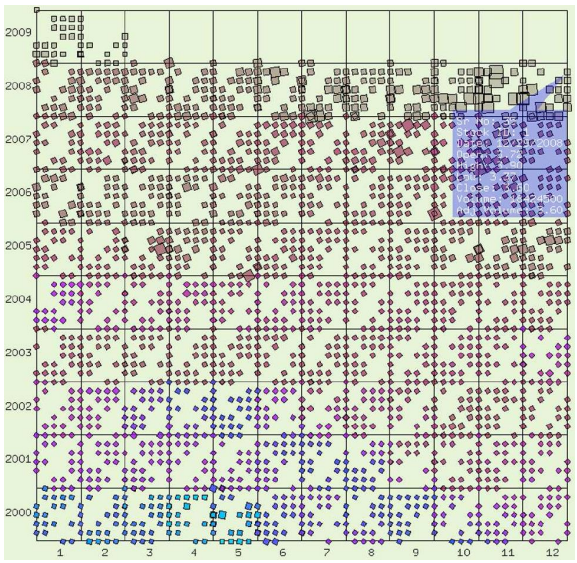


Fig. 6. Scatterplot example for Geometrically-transformed displays. Stock trading information (x-axis: month, y-axis: year) is presented by different colors, orientations, and sizes.

investment plan experiments [38]. In the experiments, risk information is presented with line graphs to evaluate if investors are aided to understand investment risks with the line graphs. In order to alleviate the overplotting issue, Lin et al. [36] adapt the fish-eye technique [64] in exploring currency exchange data streams divided by areas of interest.

Visualizations other than line graphs are also utilized. Dao et al. [22] utilize wedge charts as an extension of pie charts. In the wedge charts, each wedge is assigned to a stock, and the size and color of a wedge are mapped to velocity and force indicators in the market. 2D maps have been extensively used in information visualization, but only a few systems in the financial data analysis domain exist that project financial data onto maps. Examples of such systems and visualizations include the visualizations for comparing regional sales competition [4], regional fund distributions of market values [65], and changes in commodities markets [66].

In summary, 2D displays are used for presenting time-series information in financial data, mainly with stock data. But the 2D displays in general are not equipped with other techniques for the overcoming overplotting issue except the work by Lin et al. [36] that utilizes the fish-eye technique. We think that the overplotting issue partially leads to the adaptation of other visualizations techniques for financial analysis and leaves line graphs in auxiliary views.

7.1.2 3D Displays

Adding another dimension extends 2D visualizations to 3D visualizations. A great deal of research has demonstrated that using 3D visualizations could enhance understanding of data compared to 2D visualizations, and various 3D approaches have been adapted for financial data analysis. A standard way is to reuse known 2D visualizations. Bar (or column) charts have been popular for the extension of the standard charts (e.g., analysis for stock [67], trading transactions [33], funds [20], and customer data [42]). The basic approach used in this type of reuse is shown in Dwyer and Eades [51], whose visualization displays a navigable three-dimensional map with different sizes of columns presenting prices and volumes. Extending line graphs to 3D generates a surface chart [34], [35] for analyzing stock prices. But the surfaces still may produce occlusion and cause difficulty in user understanding. For example, line graphs for presenting price-time information are extended in [34], but a large surface positioned at the front could hide other surfaces behind the front surface in the example.

Advanced visualizations have also been extended for 3D visualizations. Examples include 3D scatter plots, splat visualizations for analyzing fund composition [65], and animating time-added 3D wedge charts for examining changes in inflation data through time [30]. Using spheres as a basis for organizing visualization is a popular approach, because they support intuitive navigation and utilization of gestalt psychology [68], and produce visually-pleasing visualizations (e.g., [24], [29]). For example, in Brath and MacMurchy's work [24], SphereCorr spreads a dense graph for presenting stock data on a sphere based on weights of links, while SphereTree presents a hierarchy of stocks, consumer price index, and occupation and incomes with treemaps on a sphere. However, they also reported issues of depth perception and limited navigation methods.

We observe that the approach of using and extending the standard charts are still limited to present time-series data such as stock market data. Even though the approach has strength in certain tasks (e.g., rank and associate [12]), they lack a method to overcome visual clutter as the size of financial become large. One effective approach is to use them in multiple coordinate views where the information with the standard charts can complement the information from other views. In our review, we find that only 14 visualizations incorporate multiple views to support financial data analysis.

7.2 Geometrically-transformed displays

Geometrically-transformed displays help users find interesting transformations of multidimensional datasets [44]. In this category, scatterplot matrices, parallel coordinates, and cluster visualizations are included. It is notable that many visualization techniques in this category are often combined with automated algorithms. Conventionally, a scatterplot presents data as points in a 2D Cartesian space. A popular approach in using the scatterplot is to change color, size, shape, and orientation of the points

with stock data [26], [69], [70]. For example, stocks are placed based on monthly (x-axis) and yearly (y-axis) trading records, and each stock is assigned to a color to present a degree of the price as shown in Figure 6. Often, the scatterplot is combined with automated algorithms. Two research projects [32], [52] use Multi Dimensional Scaling in order to analyze trading data [32] and correlations of stock markets [52]. K-core (a family of k-means) is also utilized with a scatterplot to cluster stocks based on stock price variation in order to identify barriers and resilience in stock trading [48]. Note that stock data are mainly utilized with scatterplots due to the fact that scatterplots assign properties (e.g., color, size) to individual elements and stock information is usually analyzed at the individual stock level.

On the other hand, parallel coordinate plots have not gained popularity for financial data analysis. The work of Alsakran et al. [71] is the only visualization utilizing the tile-based parallel coordinate plot for mutual fund data analysis. The system performs three processes in order to present a density of stock prices. The system first divides a visualization space by rectangular tiles. Then, the number of polylines (presenting stock prices) that passes each tile is used for computing density on each tile. By assigning different colors and opacity levels on the tiles according to the computed density, the visualization presents density of stock prices in a form of parallel coordinate plot. Note that parallel coordinates are rarely utilized for financial visualizations. This implies that multidimensional financial data have not been highlighted much yet, even though there exist several multidimensional financial data sets such as marketing data (e.g., customer data).

As discussed in Section 6, Self-Organizing Maps (SOM) generate both 2D projected data and clusters. We place cluster visualizations based on SOM in this section because SOM affects the geometry of the data and generates a new topology. There are systems using the generated topological clusters generated that are laid out in a hexagonal grid in order to analyze companies' financial performances (e.g., operating margin) [72], debt indicators [31], and a currency crisis model [37]. SOM is one of the unsupervised algorithms. This implies that SOM may produce uninteresting discoveries due to the lack of directions in learning. To complement this issue, Schreck et al. [23] present an interactive visualization framework enabling combination of (unsupervised) automated algorithms and human supervision for clustering and visualizing trajectories of stock data.

7.3 Iconic displays

Icon and glyph-based displays assign a dimension to a feature (e.g., shape, color) of the icon. Chiu et al.'s [73] star-glyph follows this idea by placing each evaluation factor (e.g., size, return of assets) for each dimension. Their visualization allows efficient comparison of strengths and weaknesses among firms. Sawant [70] also uses this idea in their visualization technique called "Psquares" (perceptual squares) when exploring stock price data where attributes of data are mapped to squares that vary in color, size, orientation, and transparency. These are then presented in a spiral layout or a scatter plot and help understanding stock trading patterns based on different time ranges (monthly or yearly). However, the large number of encodings mapped for the multiple attributes can make it difficult for analysts to discern patterns inherent in the data.

Lei and Zhang [74] use the notion of visual signatures to map financial time series datasets. They create a galaxy spiral visualization to present the market index structure and another spiral visualization technique for displaying single stock data that allows investors to group stocks with similar short-term trading activities. Sorenson et al. [2] also utilize a visual mapping technique to describe discrete event data (e.g., news, earnings, and announcements) to alphanumeric pictographics on line graphs, as opposed to using conventional abstract glyphs. However, there are usability issues in these mapping-based visualizations. The visual signature approach [74] gains lower scores than RingMap [48] in terms of usefulness, ease of use, and ease of learning. In the pictographics approach [2] there is a fixed number for mapping (36 glyphs) and occlusion issues. Shen and Eades [75] provide an ambient information visualization technique that maps financial data to an actual 3D tree where the trunk of the tree represents trade volumes and leaves show price. However, the focus of their work is to show the general changes in financial data in a visually-pleasing artwork rather than to support actual analysis.

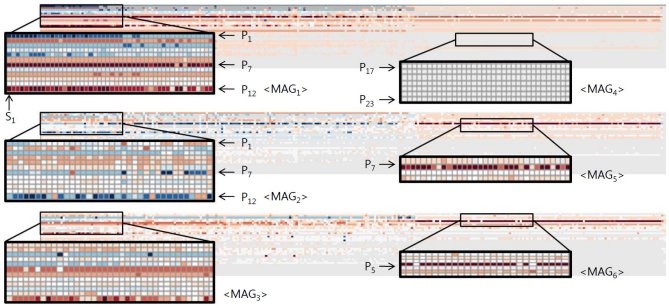


Fig. 7. An example of dense displays allowing analysis of sales, trend, and growth rates in 288 stores simultaneously [4]. Dense displays have an advantage in presenting large data.

7.4 Dense pixel displays

Dense displays (or pixel-oriented/pixel-based displays) map an attribute value to a color-coded pixel in order to best utilize the screen space. The fundamental difference among the different types of dense displays is the granularity that usually determines a layout of pixels. The standard layout strategy in this category is a grid layout as shown in the work by Ziegler et al. [76], [77]. In the work, they compute performance and growth of different financial assets (bond and fund) for all possible time intervals and present them in a single dense view. Their dense matrix visualizations enable users to compare more than 12,000 assets simultaneously. In a similar manner, Schaefer et al. [63]'s visualization allows the analysis of 222 stocks over 5 years in the oil sector. Lastly, the Pixel Bar Charts technique [78] transforms a line graph into a bar graph densely filled with color-coded vertical lines. A series of these charts are utilized to analyze market stability and volatility [43].

Dense displays have been well adapted for various sizes of pixels and layouts. Examples of systems adopting larger or varying-size of pixels include MarketAnalyzer [4], WireVis, and Alsakran et al.'s [79] system. MarketAnalyer, as shown in Figure 7, maps different colors to the same size pixels for allowing competitive advantage analysis among 288 stores based on sales, trends in the sales, and growth data. Compared to MarketAnalyzer, WireVis and Alsakran et al.'s [79] system generates larger pixels to help analysts detect suspicious activities in wire transfer transaction data [3] and find patterns in mutual fund based on various

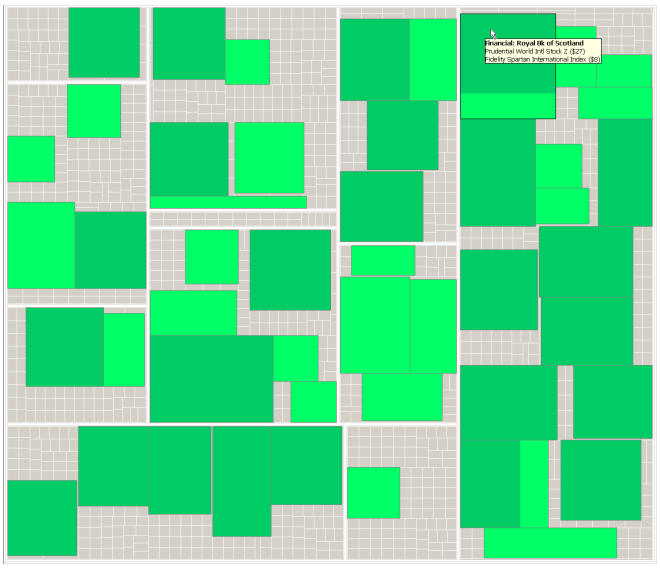


Fig. 8. Stacked displays have been popular in visualizing stock and fund data. Here, stocks included by multiple funds are presented by rectangles [82].

factors such as return against fund asset size (or cash holding, expense ratio). When the size of pixels becomes larger, the pixel-oriented visualizations tend to be more similar to heatmap visualizations [80].

Dense displays are versatile because they are easily utilized in a number of layouts. One popular layout, other than grid for the pixel-oriented display, is the radial that is advantageous in finding cyclic patterns. "Circle Segments" [25] is the first work where dimensions of stock data are mapped to segments of a circle. This approach is sufficiently scalable compared to the line graphs for a large stock price data in that it allows analysis of 265,000 data items in 50 dimensions. In a similar way, Ringmaps [48] are constructed with a varying size of multiple rings, each of which presents multiple stocks within a close data range.

Small multiples are a visualization technique where a series of basic graphics or charts on the same scale are aligned for allowing easy comparisons [81]. There are two factors that need to be determined for using this technique: representation (e.g., line graphs) and an arrangement method (e.g., grid or radial). A basic approach utilized for analyzing stock data is to map data to representative charts (e.g., line chart) by using various layout algorithms (e.g., distance based mapping [19], trajectory bundling [21]). The charts are then distributed and organized in a topological order. There are projects [19], [21], [23] that use a line chart as a basic representation combined with SOM for the chart arrangement for stock data analysis. Even though techniques for small multiples tend to be the standard 2D charts, we place small multiples in this category because much of the work utilizes the technique in presenting large data for financial data analysis.

7.5 Stacked displays

Stacked displays present data in a hierarchical manner. Treemaps [83] are a popular technique in this category whose goal is to visualize the hierarchy of multidimensional data. In treemaps, a dimension (e.g., a market sector) is connected to a rectangle whose properties (e.g., size, color, label, position) are determined by the connected dimension. This approach has been shown to have advantages for categorization and rank [12], but also has limitations such as lack of capability for presenting evolution of market or asset prices over longer time periods and a perceptional difficulty (i.e., hard to notice small size data in treemaps).

The strength in presenting hierarchies (or sectors) generates high demand in stock data analysis for providing overview stock markets [28], [40], [82], [84] where the volume, capital size, or risk of each sector determines the size of each rectangle as shown in Figure 8. Treemaps can also be used for computing layouts for placing financial data. Hao et al. [85] demonstrate how to place visualization areas (rectangles) to present sales data with bar charts. For 3D extension, two approaches are used. The first method directly extends the 2D treemaps for industry sector analysis into 3D treemaps for visual financial surveillance by adding stock prices as height information [86]. Another approach is to project treemaps onto a 3D sphere as Brath and MacMurchy [24] demonstrate for stock market analyses. However, both 3D approaches have a usability issue: occlusion and difficulty in data navigation. In the approach using additional data mapped to height, a cube with low height tends to be hidden by large surrounding cubes, and in the 3D sphere approach, users are frequently asked to perform rotation of the sphere to verify assets in both foreground and background.

7.6 Analysis of Visualization Techniques

We find that researchers have traditionally used standard charts (2D and 3D) for financial visualizations (22 out of 50 papers). We also observe that geometrically-transformed displays (10) and dense displays (13) are more preferred than iconic and stacked displays. On the other hand, we see different usage patterns over the years. The standard charts in 3D and stacked visualizations were used in early financial visualization systems followed by the standard 2D charts. Then, from 2007, dense displays including pixel-oriented visualizations gained popularity, followed by geometrically-transformed displays from 2009. We believe that a possible reason of this trend is the appearance of big data that is hard to visualize using conventional techniques and could be best utilized with consideration of data mining techniques (for reduction of data size), geometrical-transformation, and layout algorithms.

Geometrically-transformed displays (6) and standard charts (6) are equally preferred by industry. We believe that the popularity of using transformation techniques implies that financial datasets tend to be large, multidimensional, and hierarchical. Other than the standard charts and geometrically-transformed displays, the other categories have a similar number of publications that are involved with industry experts.

8 Interaction Methods

Various interaction methods have been developed in order to support the different data types and analysis methods to enable users to effectively explore their data. In order to categorize the various interaction methods, a number of surveys are presented with a focus on low level interaction operations [16], tasks [17], and benefits and user intentions [87]. In this work, we utilize Yi et al.'s [87] seven categories of interaction for the categorization of the interaction methods used in financial visualization systems. Note that not all the papers we reviewed described interaction methods in their visualizations or systems. Therefore, we exclude

	1992	1997	2003			2004	2005			2007			2008		2009	2010			2011		2012		2013	2014		
	Jungmeister92 [19]	Brodbeck97 [22]	Csallner03 [29]	Simunic03 [30]	Taskakya03 [31]	Dwyer04 [32]	Hao05 [35]	Lin05 [36]	Smeulders05 [37]	Chang07 [3]	Schreck07 [41]	Dao08 [46]	Tekusova08a [47]	Ziegler08 [48]	Schreck09 [53]	Alsakran10 [54]	Lei10 [55]	Ziegler10 [57]	Lei11 [59]	Schaefer11 [63]	Brath12 [64]	Ko12 [4]	Sorenson13 [2]	Lemieux14 [65]	Sum	From Industry
Select		■											■		■		■				■	■	■		7	4
Explore		■																							1	1
Reconfigure						■			■				■								■	■			5	1
Encode																			■						1	0
Abstract	■	■		■		■	■	■		■			■			■	■	■		■		■			13	4
Filter		■	■		■		■				■	■			■			■		■		■		■	11	2
Connect										■												■	■		3	2

TABLE 6
Categories for interaction methods. Abstract, Filter and Connect have been the most popular interaction methods for visual financial analysis. The popularity of Connect implies wide adoption of multiple coordinated views.

the papers from this list if enough information on the interaction methods cannot be discerned. Next, we briefly present the interaction taxonomy we use and examples that we considered. These interaction methods consist of: *select*, *explore*, *reconfigure*, *encode*, *abstract/elaborate*, *filter*, and *connect*.

The "Select" interaction method enables users to *mark a data in order to keep track* even when representations are changed. In order to make items visually distinctive, users click on individual items or draw an area for inclusion of items in the area. The basic result of this interaction is to highlight specific items (e.g., stocks) and provide more information for the selected item on a (pop-up) tooltip (e.g., [2], [4], [24], [32], [48]). This step is sometimes also used as a pre-step for another process (e.g., changing data ranges and hierarchy levels [30], allowing users to edit on stock trajectory for training an automated algorithm [23]).

The "Explore" interaction method allows users to *examine different subsets of the data*. The most common example of this interaction type is panning that refers to the movement of a camera (e.g., panning on maps). In financial data analysis, this interaction method is rarely used because most visualizations present data according to a given screen space. The exception is Brodbeck et al.'s work [32] where data items are scattered beyond a given screen space.

The "Reconfigure" interaction method allows users to *view data from different perspectives*. The most common implementation of reconfigure is rotation in 3D visualizations to avoid occlusion (e.g., [20], [24], [30]). In particular, Brath and MacMurchy [24] project stock data onto a sphere and allow users to rotate the sphere to view the data that are at the back of the sphere. Sorting and other rearrangements of visual items are another example in this category. Two examples that explicitly rearrange the items include sorting the customer data for developing marketing strategies [42], and reordering of stores and products based on sales performance [4] that allow quick comparisons among data items.

The "Encode" interaction corresponds to *fundamental changes in visual representations*. Examples include changes in color, size, font, orientation, and shape of the different objects (e.g., from a pie chart to a histogram). Lei and Zhang [74] present the only system that explicitly supports this interaction. In their visualization, the size of dots is allowed to be changed by users in order to encode different sizes of capitalization of industries.

The "Abstract/Elaborate" interaction allows users to change the abstraction level in the visual representation. More specifically, with support of this interaction, users are able to get both an overview and details of data items as needed. Zooming is an example interaction method in this category, and we find various usages of zooming in data exploration. When the data presented on the screen are too crowded, users are allowed to zoom-in to focus more on individual data items. Visualizations that adapt dense-displays for large data usually provide this interaction (e.g., [32], [48], [63], [71] for stock market data analysis). In the same context, this interaction is provided as a means to overcome visual clutter generated in the standard charts (e.g., standard charts for stock market data [19], [36], consumer price index [30] and fund [20] analysis). Additionally, this interaction also enables users to analyze data in different hierarchies that are generated by different types of aggregation. For example, stock market data are often aggregated by various hierarchical industry sector levels (e.g., Jungmeister and Turo [84] and Hao et al. [85] who utilize treemap visualizations to visualize hierarchical data). When data are clustered, a hierarchy is formed that can be utilized for stock pattern recognition as shown in [43]. Data aggregation by date, time, or industry sectors also form a hierarchy. WireVis [3] supports analysis of data under different temporal hierarchies.

The "Filter" interaction helps users to *analyze data with specific conditions*. We find two approaches for specifying the conditions. The first approach is to select individual objects or attributes of the objects. This approach is realized by filtering using check boxes, or clicking on an object or attribute. The object-based filtering method is mostly used in stock analysis where individual stocks need to be filtered (e.g., [21], [22], [43], [82], [85]), while attribute-based options are used when auxiliary data are used together (e.g., stock analysis with new stories [27], income trading analysis with customer data [32]). The second approach for filtering is based on range. In general, this approach is realized by providing slider bars for range selection. For example, for stock analysis, slider bars have been utilized for selecting performance range [63] and also to allow modification of relevance functions to emphasize specific regions in the x- and y-axes [77].

The "Connect" interaction means that data items are *highlighted and associated*. In general, systems with multiple coordinated views are assumed to provide this interaction, but we find only three papers in our review that utilize this interaction. For this interaction, users can click and select a region to select the data in one or more views. In MarketAnalyzer, a system designed for analyzing sales data [4], users can draw a rectangle in order to select stores and products. This action further aggregates sales for the selected stores and products in other linked views. Similarly, the WireVis system [3] designed for transaction analysis, allows brushing and linking using the mouse that are reflected in all other views including the main heatmap view. Finally, Sorenson and Brath [2] utilize the mouse hover functionality to help users determine the dates of important events using linked views.

We find from our survey that the Abstract and Filter interactions have been the most popular interaction methods in the visual analytics systems for financial data. We also find that the Select interaction has gained popularity after 2008 and sometimes it is combined with Connect. This implies that the brushing and linking technique [44] is gaining popularity. On the other hand, we find that the Explore, Reconfigure, Encode, and Connect interaction methods have not been utilized much for financial data analysis. One reason could be that we are strict in admitting that an interaction method is implemented in a system. Unless a

	1992	1995	1996	1997	1998	1999	2000	2002	2003			2004		2005		2006	2007							2008				2009			2010				2011					2012		2013	2014	Sum	From Industry
	Jungmeister92 [19]	Strausfeld95 [20]	Ankerst96 [21]	Brodbeck97 [22]	Kirkland98 [23]	Gresh99 [24]	Wattenberg99 [25]	Groenen00 [26]	Eklund02 [27]	Xiong02 [28]	Csallner03 [29]	Simunic03 [30]	Taskaya03 [31]	Dwyer04 [32]	Nesbitt04 [33]	Shen04 [34]	Hao05 [35]	Lin05 [36]	Smeulders05 [37]	Keim06 [38]	Schreck06 [39]	Chang07 [3]	Mirel07 [40]	Schreck07 [41]	Tekusova07 [42]	Ziegler07 [43]	Ziegler07a [44]	Chiu08 [45]	Dao08 [46]	Tekusova08a [47]	Ziegler08 [48]	Alsakran09 [49]	Huang09 [50]	Rudolph09 [51]	Sawant09 [52]	Schreck09 [53]	Alsakran10 [54]	Lei10 [55]	Wu10 [56]	Ziegler10 [57]	Gross11 [58]	Lei11 [59]	Sarlin11a [60]	Sarlin11b [61]	Savikhin11 [62]	Schaefer11 [63]	Brath12 [64]	Ko12 [4]	Sorenson13 [2]	Lemieux14 [65]		
Use Case	■	■	■	■	■		■	■	■	■	■	■		■	■	■		■	■	■		■	■	■	■	■	■	■	■	■		■	■	■	■	■	■	■	■	■		■	■	■	■	33	10
Case Study						■		■					■		■		■					■									■	■				■	■				■	■			■			13	7
User Feedback														■											■					■							■					■			■	■		■	8	2
User Study															■																			■			■											4	1
Performance																	■				■																											2	0

TABLE 7
Categories for evaluation methods. Use cases were mostly preferred for evaluation of visualizations or systems followed by case studies. In contrast, user study and evaluation by performance were not popular.

clear description for an interaction method is provided, we do not acknowledge a system supports the interaction method. But, we think that it is also possible that the descriptions for the interaction methods might not be provided in the paper, even though the interactions are supported. For example, 3D visualizations are a good candidate of Reconfigure. However, we found only three papers [20], [24], [42] that provide information on the interaction methods they support. In the same context, we find only 3 systems providing Connect in a system, but actually there are 15 systems incorporating multiple visualization views in systems. When we started our survey, we expected that many financial systems would provide sorting techniques (Reconfigure) as shown in Merino et al.'s list of "type of action" [12]. However, the number of systems that support sorting is much less than our expectations. Similarly, the number of systems that provide the Connect interaction is also lower than our expectations. Furthermore, we note that the systems that incorporate spatial maps for data exploration usually support the panning interaction (Explore). But we have been unable to find many papers that adopt maps for financial data analysis. Ko et al. [4] provide a map in their system to show aggregated sales results, but do not allow panning. This indicates that spatial data visualizations have not been extensively utilized for financial data analysis, even though many business domains continuously generate spatial data (e.g., marketing analysis, customer analysis, fleet/shipping administration).

9 Evaluations

Evaluating the performance of proposed visualizations or systems is an important aspect that needs to be considered before actual deployment of a system. In order to evaluate visualization techniques, Merino et al. [12] perform empirical performance experiments to validate the suitability of visualizations for different type of actions (e.g., identify, locate, distinguish, categorize, cluster, rank, compare, associate, and correlate). Marghescu [13], on the other hand, focuses on the evaluation of visualization techniques for multidimensional data with low level business and data mining questions, including outlier detection, clustering description, and comparison of items. Compared to the two previous approaches, Jeong et al. [88] select a different approach to evaluate a visual analytics system. They focus on the analysis of user interaction that was recorded from their user study with the Wirevis system [3]. Their efforts are unique because they design a new visualization system whose sole purpose is to analyze the recorded interaction data. With the new system, they clarify the relationship between user interaction and financial visual analysis.

During our survey, we identify different types of evaluations methods that were utilized in visual analytics systems for financial data and categorize them into five different categories. These are: use case, case study, user feedback, user study, and performance. A summary of our review has been provided in Table 7. When a paper presents a use case, it generally focuses on describing a hypothetical use case scenario using the proposed technique or system. On the other hand, when a case study is used for evaluation, our criterion is that there are domain experts who use the system with domain-specific data sets and present detailed information on the results produced during analysis. We count a case study in a paper if at least one author is from industry or the paper explicitly mentions that external experts are involved in the evaluation process. If not, a paper is categorized as a use case.

Compared to a use case and a case study, a user study for evaluation is different in that there are actual participants (with no previous background of the proposed visualization or system) who use the system to solve questions designed to evaluate features of the systems (e.g., user studies with the surface visualization designed for stock trading pattern analysis [34], RingMap [48] designed for stock market analysis) or to verify human response based on academic models (e.g., observing human decision-making processes in investments based on economic risk models [38], [39]).

Sometimes qualitative feedback from users in user studies or domain experts in case studies are provided, discussing pros and cons of a visualization or system. However, qualitative feedback alone cannot be a reliable indicator on usefulness of a proposed visualization or system, because personal preference is not equal to usefulness and also there are many factors (e.g., appearance of a visualization) that may cause biases on a visualization. Finally, the performance category means that authors run their algorithms and compare results with other known standard methods. We find two papers that use this evaluation technique [85], [89]. Both of these works present methods for placing stock data, but different factors are evaluated, e.g., space efficiency in [85] and normalized ordering error in [89].

We find from Table 7 that use cases are the most popular method for evaluation (33 out of 50), followed by case studies (13 out of 50). As expected, it is rare that qualitative feedback alone is provided in order to evaluate a system. Such feedback is provided mainly along with use cases. We also confirm that

not many techniques are tested using the performance testing strategy. There are two papers published in 2005 and 2006 that conducted their evaluation by measuring performance in order to show usefulness of proposed algorithms for real-world data sets. We note that performance testing may raise issues during analysis as memory and other performance issues will need to be factored in as well.

10 Discussion

Financial data analysis is important in that it directly impacts assets and investments. Even though there have been many visualization systems for analyzing the financial data, financial market participants still use conventional visualization techniques in their every day analysis. Why state-of-the-art visualization techniques have not been disseminated could be explained based on sparse collaboration between researcher and market participants; researchers cannot access real-world financial data, whereas analysts and investors have hard time finding more suitable visualization techniques than what has been used for their analysis. To mitigate this, our work focuses on two aspects; 1) we interviewed a number of analysts of various backgrounds in industry in order to clarify what analysts think is important in visualizations and analysis systems and what functionalities are needed for their tasks, 2) we surveyed the visualization and visual analytics systems allowing the financial data exploration in order to enhance the understanding of researchers, analysts, and investors on visual financial data exploration systems. Next we summarize, discuss, and derive insights.

There are 16 confirmed financial domains (Table 1), but not many domains have been beneficial from visualizations and visual analytics systems. We think that based on our survey, the work from the visual analytics community can cover up to 9 business domains (e.g., financial risk management, fraud/surveillance, economic analysis, capital management, and portfolio management). In addition, most research on visualizations and systems concentrates on stock and fund data. This can be explained in terms of easy access of real-world data and the number of stakeholders. In contrast, not much research and collaboration have been performed for analyzing companies' performance. This is expected because such analysis could be involved with confidential data (e.g., sales). Therefore, many business domains remain underutilized that otherwise could inspire new visualization techniques for visual analytics communities.

There are three categories in a broad sense in utilizing automated techniques: data reduction, clustering, and forecasting. Considering the fact that not many automated techniques are applied to financial data in the past, we think that there are still many opportunities for research in this aspect. The analysts we interviewed actually stress that many automated algorithms are applied to their data in everyday analysis that are not available in the current visual analysis tools including Tableau and Spotfire. On the other hand, we think that it is possible that this narrow use of the automated techniques might be caused by the use of a single type of data– time-series. In this sense, applying different automated techniques to various data types could inspire new insights in data and visualizations. For example, the analysts we interviewed stress importance of the role of automated anomaly detection with multiple data sources. In order to provide accurate anomaly detection results, one can combine financial data with text data generated in various fields and apply event detection methods [90].

There is preference in utilizing automated techniques for reduction (data size and dimensionality). We conjecture that the advent of big data boosts adoption of automated techniques in visual analysis systems in order to reduce data size and produce clusters. In the same context, visualizations that are able to present more (processed) data will gain popularity in use as geometrically transformed and dense pixel displays have been done.

We see strong preferences on the Abstract and Filter interaction methods. It is possible that this preference is caused by the data used or a type of task during analysis. We think this implies a general process in analysis. It is often to start analyzing data with higher abstraction levels in visual representations. Then, in order to move to another point of analysis, users tend to use filtering or change abstraction levels. Once a new insight is found, the data is reprocessed with automated techniques if applicable. In fact, this pattern data analysis demonstration leads us to the financial data exploration pipeline as shown in Figure 1. On the other hand, Connect does not gain much popularity, which is different from our initial expectation.

Conventionally, evaluation of visualizations and systems heavily relies on use cases. This may prevent industry experts from using proposed techniques or approaches with confidence for big data analysis. In order to overcome this, evaluation via performance experiments is needed to provide memory consumption and computation time. In addition, not many papers utilize domain experts for evaluation. This may be caused by the fact that domain experts who can bring their data for use on a system is not always possible. In this case, we think very detailed qualitative feedback of non-expert users should be pursued, not to mention the user study with questions designed for public users.

11 Conclusion

In this work, we Aave presented a Aurvey Af the work Aor vi-sual approaches Aor Axploring financial Aata. Our Aurvey Ahows that there are trends and preferences in Aata Aources, automated techniques, visualizations, interaction methods and Avaluation. In reverse, this implies that there are Atill many Andermined business Aomains where Aesearch An Aew techniques and systems are Aeeded. Ae Aelieve that the lessons Arom Aur survey Ahed Aome light An Anderstanding Atate-of-the-art visual analytics approaches Aor financial Aata analysis.

Acknowledgments

This work was supported in part by the U.S. Department of Homeland Security's VACCINE Center under award no. 2009-ST-061-CI0002.

References

[1] R. Edwards, J. Magee, and W. Bassetti, *Technical Analysis of Stock Trends*. AMACOM, 2007.

[2] E. Sorenson and R. Brath, "Financial visualization case study: Correlating financial timeseries and discrete events to support investment decisions," in *Proceedings of International Conference on Information Visualization*, 2013, pp. 232–238.

[3] R. Chang, M. Ghoniem, R. Kosara, W. Ribarsky, J. Yang, E. Suma, C. Ziemkiewicz, D. Kern, and A. Sudjianto, "Wirevis: Visualization of categorical, time-varying data from financial transactions," in *Proceedings of IEEE Conference on Visual Analytics Science and Technology*, 2007, pp. 155–162.

[4] S. Ko, R. Maciejewski, Y. Jang, and D. S. Ebert, "Marketanalyzer: An interactive visual analytics system for analyzing competitive advantage using point of sale data," *Computer Graphics Forum*, vol. 31, no. 3, pp. 1245–1254, 2012.

[5] M. Pryke, "Money's eyes: the visual preparation of financial markets," *Economy and Society*, vol. 39, no. 4, pp. 427–459, 2010.

[6] J. Schwabish, "An economist's guide to visualizing data," *Journal of Economic Perspectives*, vol. 28, no. 1, pp. 209–234, 2014.

[7] M. Flood, V. Lemieux, M. Varga, and B. Wong, "The application of visual analytics to financial stability monitoring," *Social Science Research Network, SSRN Scholarly Paper ID 2438194*, May 2014.

[8] D. P. Tegarden, "Business information visualization," *Communications of the Association for Information Systems*, vol. 1, no. 4, pp. 197–204, 1999.

[9] J. J. Thomas and K. A. Cook, Eds., *Illuminating the Path: The R&D Agenda for Visual Analytics*. IEEE Press, 2005.

[10] M. Dumas, M. J. McGuffin, and V. L. Lemieux, "Financevis.net - a visual survey of financial data visualizations," in *Poster Abstracts of IEEE Conference on Visualization*, 2014.

[11] "FinanceVisBrowser," http://financevis.net/. Accessed by 15 Feb 2015.

[12] C. S. Merino, M. Sips, D. A. Keim, C. Panse, and R. Spence, "Task-at-hand interface for change detection in stock market data," in *Proceedings of ACM Conference on Advanced Visual Interfaces*, 2006, pp. 420–427.

[13] D. Marghescu, "Multidimensional data visualization techniques for financial performance data: A review," Turku Centre for Computer Science, Tech. Rep., 2007.

[14] D. A. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann, *Mastering The Information Age-Solving Problems with Visual Analytics*. Florian Mansmann, 2010.

[15] P. Pirolli and S. Card, "The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis," pp. 2–4, 2005.

[16] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *Proceedings of IEEE Symposium on Visual Languages*, 1996, pp. 336–343.

[17] R. A. Amar, J. Eagan, and J. T. Stasko, "Low-level components of analytic activity in information visualization," in *Proceedings of IEEE Symposium on Information Visualization*, 2005, pp. 111–117.

[18] R. Kohavi, N. Rothleder, and E. Simoudis, "Emerging trends in business analytics," *ACM Communications*, vol. 45, no. 8, pp. 45–48, 2002.

[19] K. Simunic, "Visualization of stock market charts," in *Proceedings of International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, 2003.

[20] T. Dwyer and D. R. Gallagher, "Visualising changes in fund manager holdings in two and a half-dimensions," *Information Visualization*, vol. 3, no. 4, pp. 227–244, Dec. 2004.

[21] T. Schreck, T. Tekušová, J. Kohlhammer, and D. Fellner, "Trajectory-based visual analysis of large financial time series data," *ACM SIGKDD Exploration Newsletter*, vol. 9, no. 2, pp. 30–37, 2007.

[22] H. T. Dao, A. L. Bazinet, R. Berthier, and B. Shneiderman, "Nasdaq velocity and forces: An interactive visualization of activity and change." *Journal of Universal Computer Science*, vol. 14, no. 9, pp. 1391–1410, 2008.

[23] T. Schreck, J. Bernard, T. Von Landesberger, and J. Kohlhammer, "Visual cluster analysis of trajectory data with interactive kohonen maps," *Information Visualization*, vol. 8, no. 1, pp. 14–29, Jan. 2009.

[24] R. Brath and P. Macmurchy, "Sphere-based information visualization: Challenges and benefits," in *Proceedings International Conference on Information Visualization*, 2012, pp. 1–6.

[25] M. Ankerst, D. A. Keim, and H. peter Kriegel, "Circle segments: A technique for visually exploring large multidimensional data sets," in *Proceedings of IEEE Conference on Visualization, Hot Topic Session*, 1996.

[26] E. Wu and P. Phillips, "Financial markets in motion: Visualising stock price and news interactions during the 2008 global financial crisis," *Procedia Computer Science*, vol. 1, no. 1, pp. 1765–1773, 2010.

[27] T. Taskaya and K. Ahmad, "Bimodal visualisation: A financial trading case study," in *Proceedings International Conference on Information Visualization*. IEEE Computer Society, 2003, pp. 320–326.

[28] M. Wattenberg, "Visualizing the stock market," in *Proceedings of ACM CHI on Human Factors in Computing Systems, Extended Abstracts*. ACM, 1999, pp. 188–189.

[29] M. H. Gross, T. C. Sprenger, and J. Finger, "Visualizing information on a sphere," in *Proceedings of IEEE Symposium on Information Visualization*. IEEE, 1997, pp. 11–16.

[30] T. Tekusova and T. Schreck, "Visualizing time-dependent data in multivariate hierarchic plots - design and evaluation of an economic application," in *Proceedings International Conference on Information Visualization*, July 2008, pp. 143–150.

[31] P. Sarlin, "Sovereign debt monitor: A visual self-organizing maps approach," in *Proceedings of IEEE Symposium on Computational Intelligence for Financial Engineering and Economics*, 2011, pp. 1–8.

[32] D. Brodbeck, M. Chalmers, A. Lunzer, and P. Cotture, "Domesticating bead: adapting an information visualization system to a financial institution," in *Proceedings of IEEE Symposium on Information Visualization*, Oct 1997, pp. 73–80.

[33] J. D. Kirkland, T. E. Senator, J. J. Hayden, T. Dybala, H. G. Goldberg, and P. Shyr, "The nasd regulation advanced detection system (ads)," in *Proceedings of National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence*. American Association for Artificial Intelligence, 1998, pp. 1055–1062.

[34] K. Nesbitt and S. Barrass, "Finding trading patterns in stock market data," *IEEE Computer Graphics and Applications*, vol. 24, no. 5, pp. 45–55, 2004.

[35] D. Gresh, B. Rogowitz, M. Tignor, and E. Mayland, "An interactive framework for visualizing foreign currency exchange options," in *Proceedings of IEEE Conference on Visualization*, 1999, pp. 453–562.

[36] L. Lin, L. Cao, and C. Zhang, "The fish-eye visualization of foreign currency exchange data streams," in *Proceedings of Asia-Pacific Symposium on Information Visualization*. Australian Computer Society, 2005, pp. 91–96.

[37] P. Sarlin and T. Eklund, "Fuzzy clustering of the self-organizing map: Some applications on financial time series," in *Advances in Self-Organizing Maps*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2011, vol. 6731, pp. 40–50.

[38] S. Rudolph, A. Savikhin, and D. Ebert, "Finvis: Applied visual analytics for personal financial planning," in *Proceedings of IEEE Conference on Visual Analytics Science and Technology*, 2009, pp. 195–202.

[39] A. Savikhin, H. C. Lam, B. Fisher, and D. Ebert, "An experimental study of financial portfolio selection with visual analytics for decision support," in *Proceedings of Hawaii International Conference on System Sciences*, Jan 2011, pp. 1–10.

[40] V. L. Lemieux, B. W. Shieh, D. Lau, S. H. Jun, T. Dang, J. Chu, and G. Tam, "Using visual analytics to enhance data exploration and knowledge discovery in financial systemic risk analysis: The multivariate density estimator," in *Proceedings of iConference*. iSchools, 2014, pp. 649–653.

[41] L. Kahaner, *Competitive Intelligence: How to Gather Analyze and Use Information to Move Your Business to the Top*. New York, U.S.A.: Touchstone Press, 1998.

[42] R. Smeulders and A. Heijs, "Interactive visualization of high dimensional marketing data in the financial industry," in *Proceedings International Conference on Information Visualization*, 2005, pp. 814–817.

[43] H. Ziegler, M. Jenny, T. Gruse, and D. Keim, "Visual market sector analysis for financial time series data," in *Proceedings of IEEE Conference on Visual Analytics Science and Technology*, 2010, pp. 83–90.

[44] D. Keim, "Information visualization and visual data mining," *IEEE Transactions on Visualization and Computer Graphics*, vol. 8, no. 1, pp. 1–8.

[45] P. Berkhin, "Survey of clustering data mining techniques," Accrue Software, Technical report, 2002.

[46] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the 5th Berkeley Symp. on Mathematics Statistics and Probability*, 1967.

[47] T. Kohonen, "The self-organizing map," *Proceedings of IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[48] S. Lei and K. Zhang, "A visual analytics system for financial time-series data," in *Proceedings of International Symposium on Visual Information Communication*. ACM, 2010, pp. 20:1–20:9.

[49] I. Borg and P. Groenen, *Modern multidimensional scaling, theory and applications*. Springer, 1997.

[50] B. S. Everitt and G. Dunn, *Applied Multivariate Data Analysis*. Arnold, 1991.

[51] T. Dwyer and P. Eades, "Visualising a fund manager flow graph with columns and worms," in *Proceedings International Conference on Information Visualisation*, 2002, pp. 147–152.

[52] P. J. Groenen and P. H. Franses, "Visualizing time-varying correlations across stock markets," *Journal of Empirical Finance*, vol. 7, no. 2, pp. 155–172, 2000.

[53] I. Fodor, "A survey of dimension reduction techniques," Lawrence Livermore National Laboratory, Tech. Rep., 2002.

[54] T. W. Liao, "Clustering of time series data: A survey," *Pattern Recognition*, vol. 38, no. 11, pp. 1857–1874, 2005.

[55] A. Inselberg, "The plane with parallel coordinates," *The Visual Computer*, vol. 1, no. 2, pp. 69–91, 1985.

[56] M. Wattenburg, "Visualizing the stock market," in *Proceedings of ACM CHI Conference on Human Factors in Computing Systems, Extended Abstracts*, 1999, pp. 188–189.

[57] N. Elmqvist, P. Dragicevic, and J.-D. Fekete, "Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1539–1148, 2008.

[58] H. Chernoff, "Using faces to represent points in *k*–dimensional space graphically," *Journal of the American Statistical Association*, vol. 68, no. 342, pp. 361–368, 1973.

[59] S. Havre, B. Hetzler, and L. Nowell, "Themeriver: visualizing theme changes over time," in *Proceedings of IEEE Symposium on Information Visualization*. IEEE, Oct. 2000, pp. 115–124.

[60] D. A. Keim, "Designing pixel-oriented visualization techniques: Theory and applications," *IEEE Transactions on Visualization and Computer Graphics*, vol. 6, no. 1, pp. 59–78, 2000.

[61] M. O. Ward, G. G. Grinstein, and D. A. Keim, *Interactive Data Visualization - Foundations, Techniques, and Applications*. A K Peters, 2010.

[62] J. Murphy, *Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications*. New York Institute of Finance, 1999.

[63] M. Schaefer, F. Wanner, R. Kahl, L. Zhang, T. Schreck, and D. A. Keim, "A Novel Explorative Visualization Tool for Financial Time Series Data Analysis," in *Proceedings of UKVAC Workshop on Visual Analytics*, 2011.

[64] G. W. Furnas, "Generalized fisheye views," in *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*, 1986, pp. 16–23.

[65] F. Xiong, E. Prakash, and K. Ho, "E-r modeling and visualization of large mutual fund data," *Journal of Visualization*, vol. 5, no. 2, pp. 197–204, 2002.

[66] B. Mirel, P. Goldsmith, R. Brath, and B. Cort, "Visual analytics for model-based policy analysis: Exploring rapid changes in commodities markets," in *Proceedings of International Conference on Digital Government Research: Bridging Disciplines & Domains*. Digital Government Society of North America, 2007, pp. 312–313.

[67] L. Strausfeld, "Financial viewpoints: Using point-of-view to enable understanding of information," in *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*. ACM, 1995, pp. 208–209.

[68] K. Koffka, *Principles of Gestalt Psychology*. Routledge, 2013.

[69] T. Tekusova and J. Kohlhammer, "Applying animation to the visual analysis of financial time-dependent data," in *Proceedings International Conference on Information Visualization*, 2007, pp. 101–108.

[70] A. Sawant, "Stockviz: Analyzing the trend of stocks in major auto, oil, consumer, and technology companies," in *Proceedings of International Conference on Modeling, Simulation & Visualization Methods*, 2009, pp. 278–284.

[71] J. Alsakran, Y. Zhao, and X. Zhao, "Tile-based parallel coordinates and its application in financial visualization," *Proceedings of International Society for Optical Engineering*, vol. 7530, pp. 753 003–753 003–12, 2010.

[72] T. Eklund, B. Back, H. Vanharanta, and A. Visa, "Assessing the feasibility of self organizing maps for data mining financial information," *European Conference on Information Systems*, pp. 528–537, 2002.

[73] T.-F. Chiu, C.-F. Hong, and Y.-T. Chiu, "Visualization of financial trends using chance discovery methods," in *Proceedings of New Frontiers in Applied Artificial Intelligence*. Springer, 2008, pp. 708–717.

[74] S. Lei and K. Zhang, "Visual signatures for financial time series," in *Proceedings of International Symposium on Visual Information Communication*. ACM, 2011, pp. 16:1–16:10.

[75] X. Shen and P. Eades, "Using moneytree to represent financial data," in *Proceedings International Conference on Information Visualization*, 2004, pp. 285–289.

[76] H. Ziegler, T. Nietzschmann, and D. Keim, "Visual exploration and discovery of atypical behavior in financial time series data using two-dimensional colormaps," in *Proceedings International Conference on Information Visualization*, July 2007, pp. 308–315.

[77] H. Ziegler, T. Nietzschmann, and D. A. Keim, "Visual analytics on the financial market: Pixel-based analysis and comparison of long-term investments," in *Proceedings International Conference on Information Visualization*, 2008, pp. 287–295.

[78] D. A. Keim, M. C. Hao, J. Ladisch, M. Hsu, and U. Dayal, "Pixel bar charts: A new technique for visualizing large multi-attribute data sets without aggregation," in *Proceedings of IEEE Symposium on Information Visualization*, 2001, pp. 113–120.

[79] J. Alsakran, Y. Zhao, and X. Zhao, "Visual analysis of mutual fund performance," in *Proceedings of International Conference on Information Visualisation*, 2009, pp. 252–259.

[80] J. Abello and F. van Ham, "Matrix zoom: A visual interface to semi-external graphs," in *Proceedings of IEEE Symposium on Information Visualization*, 2004, pp. 183–190.

[81] E. R. Tufte, *Envisioning Information*. Graphics Press, 1990.

[82] C. Csallner, M. Handte, O. Lehmann, and J. Stasko, "Fundexplorer: supporting the diversification of mutual fund portfolios using context treemaps," in *Proceedings of IEEE Symposium on Information Visualization*, Oct 2003, pp. 203–208.

[83] B. Johnson and B. Shneiderman, "Tree-maps: A space-filling approach to the visualization of hierarchical information structures," in *Proceedings of IEEE Conference on Visualization*, 1991, pp. 284–291.

[84] W.-A. Jungmeister and D. Turo, "Adapting treemaps to stock portfolio visualization," UMCP-CSD CS-TR-2996, University of Maryland, Tech. Rep., 2002.

[85] M. C. Hao, U. Dayal, D. A. Keim, and T. Schreck, "Importance-driven visualization layouts for large time series data," in *Proceedings of IEEE Symposium on Information Visualization*, 2005, pp. 203–210.

[86] M. L. Huang, J. Liang, and Q. V. Nguyen, "A visualization approach for frauds detection in financial market," in *Proceedings International Conference on Information Visualization*, July 2009, pp. 197–202.

[87] J. S. Yi, Y. ah Kang, J. T. Stasko, and J. A. Jacko, "Toward a deeper understanding of the role of interaction in information visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1224–1231, 2007.

[88] D. H. Jeong, W. Dou, H. R. Lipford, F. Stukes, R. Chang, and W. Ribarsky, "Evaluating the relationship between user interaction and financial visual analysis," in *Proceedings of IEEE Symposium on Information Visualization*. IEEE Computer Society, 2008, pp. 83–90.

[89] T. Schreck, D. Keim, and F. Mansmann, "Regular treemap layouts for visual analysis of hierarchical data," in *Proceedings of ACM Conference on Computer Graphics*, 2006, pp. 184–191.

[90] F. Wanner, A. Stoffel, D. Jckle, B. C. Kwon, A. Weiler, and D. A. Keim, "State-of-the-Art Report of Visual Analysis for Event Detection in Text Data Streams," in *EuroVis - STARs*. The Eurographics Association, 2014.

2015 48th Hawaii International Conference on System Sciences

An Integrated In-Situ Approach to Impacts from Natural Disasters on Critical Infrastructures

Sebastian Mittelstädt*, Xiaoyu Wang†, Todd Eaglin†, Dennis Thom‡, Daniel Keim*, William Tolone†, William Ribarsky†
*University of Konstanz, Email: firstname.lastname@uni-konstanz.de
†University of North Carolina at Charlotte, Email: firstname.lastname@uncc.edu
‡University of Stuttgart, Email: dennis.thom@vis.uni-stuttgart.de

***Abstract*—Natural disasters can have a devastating effect on critical infrastructures, especially in case of cascading effects among multiple infrastructures such as the electric power grid, the communication network, and the road network. While there exist detailed models for individual types of infrastructures such as electric power grids, these do not encompass the various interconnections and interdependencies to other networks. Cascading effects are hard to discover and often the root causes of problems remain unclear. In order to enable real-time situational awareness for operational risk management one needs to be aware of the broader context of events. In this paper, we present a unique visual analytics control room system that integrates the separate visualizations of the different network infrastructures with social media analysis and mobile in-situ analysis to help to monitor the critical infrastructures, detecting cascading effects, performing root cause analyses, and managing the crisis response. Both the social media analysis and the mobile in-situ analysis are important components for an effective understanding of the crisis and an efficient crisis response. Our system provides a mechanism for conjoining the available information of different infrastructures and social media as well as mobile in-situ analysis in order to provide unified views and analytical tools for monitoring, planning, and decision support. A realistic use case scenario based on real critical infrastructures as well as our qualitative study with crisis managers shows the potential of our approach.**

I. Introduction

Responding to the destructive impact of a volatile hurricane to a network of critical infrastructure is the central challenge for emergency responders. After witnessing the devastating destruction from Hurricane Sandy during 2012 and the flood disaster in Germany 2013 decision makers are on high-alert for threats to their critical infrastructures such as power lines, food networks, shelters, etc., potentially caused by impact from natural catastrophes. Important backbones of our society are electrical power networks since the electricity supply has a strong impact on the fundamental societal structures such as life/health, environment, and economy. Especially, electric power systems are increasingly dependent on information and communication technology (ICT) systems as new monitoring, control, and protection functions, especially in the currently emerging Smart Grid installations. In order to deal with the increasing vulnerabilities of electric power systems, advanced ICTs, including network-based Supervisory Control and Data Acquisition (SCADA) systems or Wide Area Monitoring Systems (WAMS) have been deployed by the power industry.

Analyzing the vast amount of information from different domains is a complex analytical issue. The monitoring of the interconnections between power grids and digital networks requires the integration of several data sources. With an overview the crisis manager is able to understand and explore the crisis allowing her/him to project the future development and to make decisions. Situational awareness is important on all levels of crisis response that range from central command centers to site-commanders and boots-on-the-ground. All levels have to access the information of a crisis. They need to communicate bottom-up or top-down since crisis managers typically rely on the information of the field and first responders lack context information.

Novel public communication platforms like social media services and other Web 2.0 sources have established a completely new information channel that can help to improve situational awareness for the decision makers. Citizens affected by critical events often report vital situation related information directly to messaging services like Twitter or Facebook. They use mobile and sometimes even GPS-enabled communication devices like smartphones or tablet computers. Gathering useful information pieces from the vast amounts of random unrelated chatter poses a completely new challenge for analysis and decision support systems.

Existing tools and systems do not support the integration of information over several critical infrastructures such as power grids and the ICT networks. The monitoring and understanding of the relationship of critical infrastructures and the coordinated management of their failures is therefore one of the biggest challenges in critical infrastructure protection and crisis response. In this paper, we present a system that supports all levels of command structures and enables situational awareness for crisis response. This system was developed within a nationwide interdisciplinary project [1] running for three years with an international research collaboration with a partner project [2].

Our contribution: We present a visual analytics system that: 1) supports all levels of crisis response with specialized equipment and visualizations for control rooms and mobile devices; 2) combines multiple critical infrastructures and

1530-1605/15 $31.00 © 2015 IEEE
DOI 10.1109/HICSS.2015.136

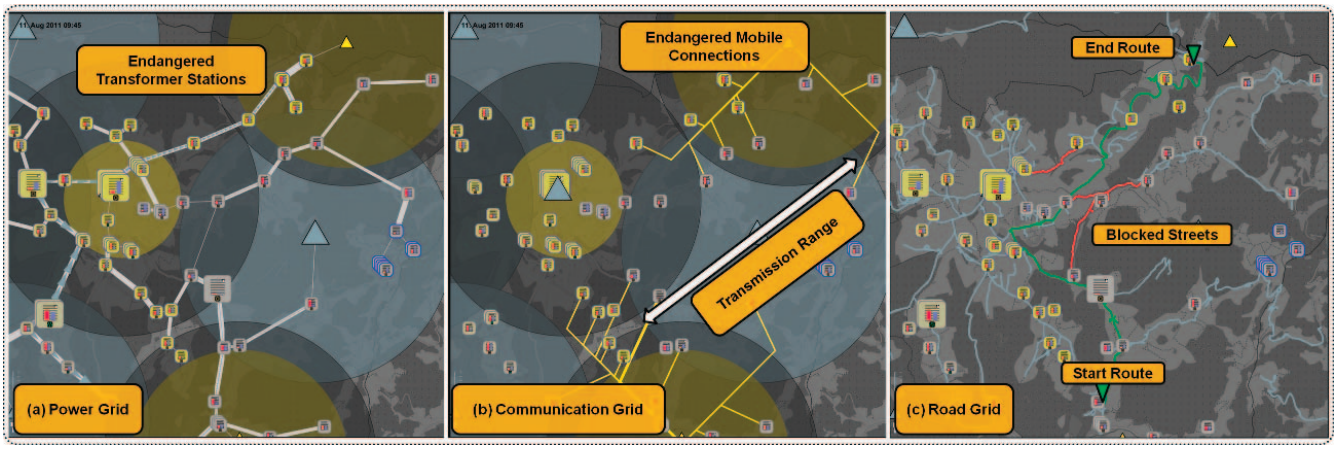


Figure 1. Overview of the power, mobile, and road grid. Transformer stations (rectangles) are connected via power lines and are also connected to the communication infrastructure (triangles), which transfers the information to the central control room. The transmission range of the mobile stations is visualized as concentric circles. While gray indicates normal operation mode, the yellow elements on the screen reveal a severe situation. High deviations in voltage cascaded from the energy grid into the mobile grid due to failures of the power supply. Now, the operator must intervene immediately. With malfunctions in the mobile grid, the crisis response commands from the control room won't reach the field, which then would result in a black out. Further, roads are blocked and hinder the response team to reach the target area, which is visible in the command center after the first responders update the street status. The dynamic routing adapts to these constraints and calculates a different route.

social media by information abstraction; 3) enables interactive simulation and visualization of the subsequent development of a crisis; 4) enables interdisciplinary and distributed teams to understand and react on crisis situations.

II. Related Work

The advantage of visual analytics has been illustrated in an analysis of the 2005 outbreak of the avian flu by combining different analysis capabilities [3]. This scenario shows the power of an analytic setting that supports the analysis of complex, real-world scenarios. Also, first visual analytics tools in the area of crisis response were the result of SoKNOS [4]. This comprehensive environment requires integrated visual and traditional systematic analysis of massive data, including improved strategies for exploratory visual analysis, hypothesis testing and user-specific presentation of relevant information as a basis for actionable decision making. Furthermore, visual analytics tools for analyzing syndromic hotspots are presented by Maciejewski et al. [5] that allow the analyst to perform real-time hypothesis testing.

Much prior research has focused on using simulation and predictive modeling to anticipate hurricane movement and suggest possible landfall and impact locations [6], [7]. Campbell and Weaver [8] investigate situational awareness during emergencies using two different tools: RimSim Response! (RSR) and RimSim Visualization (RSV). The work of Kim et al. [9] focused on the use of mobile devices for situationally aware emergency response and training, and thus, their approach is similar to our work.

As part of the PSERC project, techniques are developed to visualize complex power systems and flows [10]. GreenGrid [11] was developed to explore the planning and monitoring of the North American Electricity Infrastructure. For the interactive analysis of network related data sources such as server logs or BGP protocol data, Fischer et al. developed a visual analytics expert system in [12]. A detailed overview of cyber security and privacy issues in a smart grid context is presented in [13]. The control room is the central part of a system, where all information comes together. Notifications and alarms are collected and transferred to a central node. These events are typically evaluated by rule-based systems, where the rules are defined by domain experts. Rooney et al. gives an introduction to Root Cause Analysis (RCA) in [14]. There are also systems that use simpler fuzzy logic rules as a vehicle that allows engineers to incorporate human reasoning in the control algorithm [15]. Impacts on critical infrastructures are typically complex and involve several experts of different domains for crisis response. This demands for interactive workspaces but also for high-resolution environments that enable visualizing all elements of a crisis and their context. Approaches that utilize interactive workspaces combined with visualization devices were previously presented [16], [17].

Social media, such as Twitter and Facebook, contain time-critical information that can enhance situational awareness. First approaches for building and improving decision making systems in this domain were introduced by Tomaszewski et al. [18]. MacEachren et al. [19] developed a visual analytics tool that allows for querying social media sources and depicting aggregated results on a geographical map. Thom et al. [20] present a novel cluster analysis approach to detect spatiotemporal anomalies in Twitter messages.

The discussed approaches and systems focus on a specific domain and do not combine various external data sources. Integrate sensor data (electricity, weather, supply), social media and in-situ analysis is a challenging task. Furthermore, most of the current systems are intended for domain expert

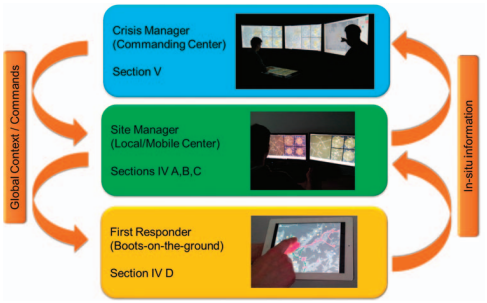


Figure 2. Overview of the Users that our system is designed to facilitate.

users, although crisis management teams may consist of interdisciplinary members. Enabling interdisciplinary teams to analyze and understand interdependent data leads to efficient crisis response.

III. Design Framework: Target Users & Requirements

Large scale emergency response has a command structure as illustrated in Figure 2, which was depicted by our crisis managers within the project. A large police or fire response, for example, will have a site commander who deploys first responders. If the emergency is larger and more wide-spread, there will be a command center with oversight over multiple site commanders. A similar structure applies to breakdown of the electrical grid or other critical infrastructures. The deeper one goes into the command structure, the more mobile the responders are; they are focused on the locales, tasks, and decisions at hand and traditionally don't have contextual understanding or situational awareness. Typically, site commanders also don't have situational awareness (in terms of the deployment of their personnel and what they are doing or seeing, for example) nor do they have the context to make the most effective decisions. To make decisions, the first responder will want to know what is happening in the locale, what is about to hit where and when. The site commander will want to know similar things over a wider locale and must in addition organize and manage a group of responders. Aspects of all this should flow up to the command center for overall decision-making.

Further, any feedback and updates they provide will be transmitted to the central system. Detailed location, movement, and action updates can be placed in the appropriate spatio-temporal context so that commanders can see, in unprecedented detail and without ambiguity, what is going on (and responders can see, minute-by-minute where there fellow responders are and what they are doing). Specifically, our system is designed to accommodate the three essential personnel in a crisis respond scenario:

Crisis Manger. Crisis Managers are a group of domain experts who oversees the entire emergence response process. This group is typically formed by interdisciplinary members from various analytical domains, such as grid operators from power companies, city-state officials, evacuation experts, and people from federal departments. Their objective is to understand and assess the severity of the crisis situation, select corresponding Site-Commanders with appropriate First Responders, and provide Just-In-Time decisions based on inputs from social media and in-field communications. The natural of the heterogeneous data inputs for the Crisis Managers determined that they will benefit from a control room setup, where they will be provided with a combined overview of the crisis situation. As detailed in section 5, our setup supports distributed as well as collaborative analysis, provides overviews of the development of the ongoing crisis, and it further enables the Crisis Managers to interactively deploy and arrange response effort, and receive the updates from site commanders in real-time.

Site Commanders. With the advanced mobile technology (e.g., mobile emergency response vehicles), site managers are the critical link between Crisis-Manager at control center and First Responders in the field. Based on our previous collaboration with local emergency responders, Site Commander (e.g., Police Chief) are often stationed near the crisis center where first responders were deployed to conduct on-site instructions and in-situ communications. At the mean time, they are relying on the mobility of the technologies to maintain an open communication channel with control center for further situation assessment and updates. Site Commanders act differently from Crisis Managers in a way that they have a more focused missions in a specific area that is assigned to them (e.g., a specific substation or a street blocks) and are in charge of provide real-time response as the crisis unfolds in the field.

First Responders. First Responders are the group that fights the diseases right in the center of where crisis occurs. These teams consist of various professionals, such as policeman, fire fighter, and power grid responders. These interdisciplinary group mainly conduct response effort in the field with instructions from Site Commanders. Their extreme needs of mobility determined that we need to provide them with a mobile-device based visual analytics system. Key functions in this system, as detailed in section 4.4, includes instructions that informs them about the areas that they need to focus their attentions, interactive methods to depict areas to prioritize their tasks, and finally communication methods to provide updates and situation reports to their Site Managers. All this information need to be shared through wireless networks that directly feedback to the Site Commanders and further to Crisis Managers.

IV. System Components

A. Simulation of Critical Infrastructures

Large scale natural disaster, a cyber-attack, or other wide spread crisis may affect multiple infrastructures. To capture these complex, multifarious, and dynamic effects, we utilize

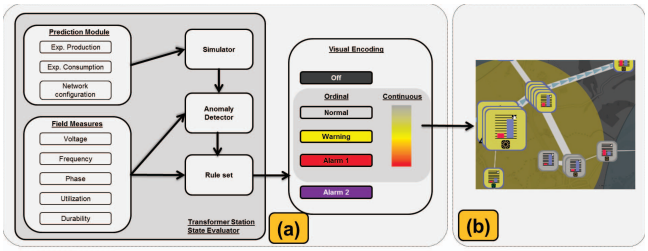


Figure 3. (a) Station state evaluators consider the incoming measurements and the comparison to the expected behavior, which reveals anomalies. A set of rules maps this input to color that expresses the status of the element. (b) Domain details are added to the symbols such as power consumption and production (red and blue bars), as well as producer types (photovoltaic or biogas) for transformer stations.

a simulation model that takes into account the interrelationships among critical infrastructures. The simulation is built within a rule-based framework for integrating multiple infrastructure components at a high level. The interlaced critical infrastructures are captures in a set of networks with each node having a set of properties according to its category and the edges providing a dependency rule according to the category and state of the two connected nodes. This results in a dependency/interdependency ontology (e.g., as illustrated in Figure 1(a) mobile transmitters are connected to transformer stations). Thus, for example, a breakdown of a power substation would immediately cascade to power loss at points on its distribution network.

In some natural catastrophes some roads may be affected and rescue or response teams, especially with heavy gear, are not able to pass them. This has to be considered in the evacuation and logistic management at all three levels: E.g., the crisis managers must plan the logistics of gear and troops; the site-commander sends in first responders on different routes to the crisis and first responders will update the status of streets if they are not passable. Our system supports dynamic routing with the state-of-the-art algorithms that consider the current status of streets.

B. Visualization of Interdependent Infrastructures

A smart grid (energy network) typically consists of power lines at different voltage levels connected by transformer stations. These stations distribute the power over regions and supply streets, households, and industrial facilities. In our scenario, a mobile communication grid transfers information and control commands from the central control room to the electrical grid. The mobile transmitters itself are power-supplied by common transformer stations and thus, the infrastructures are tightly interconnected.

1) Information Abstraction & Visual Encoding: Adapting and extending the visual abstraction presented in [21], the complex and vast amount of information of each infrastructure is reduced to the essentials, in order to enable the decision maker to understand the full crisis in its context and to detect potential cascading effects. Every infrastructure is abstracted

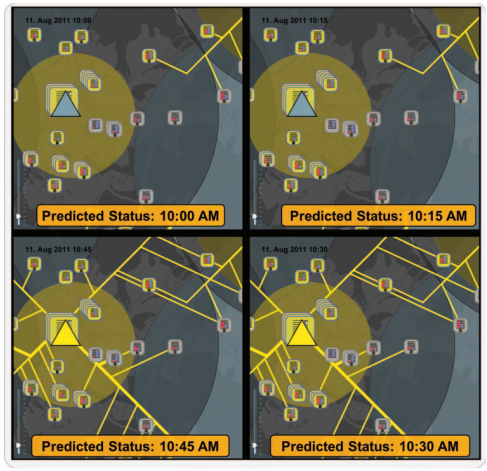


Figure 4. Subsequent Development: If there is any emerging problem in the future, the prediction view will show the future status of the network by small multiples, which shows the remaining time and the affected elements.

to an undirected graph. Its nodes are represented by symbols, such as rectangles for transformer stations and triangles for mobile transmission stations. The graph edges represent the domain dependent connection between infrastructure elements, such as power lines and mobile communication connections (see Figure 3(b) and Figure 1).

The status of each element is estimated by a state-evaluator model that is defined for each infrastructure. These models concern the actual information of the field, such as utilization and durability (see Figure 3(a)). We use a prediction module for our power grid that predicts the consumption and production at each transformer station according to weather forecasts and past data based on Monte Carlo simulation. This information is sent to the simulation server that simulates the subsequent development. This "expected" behavior is compared to the actual measurements. Thus, anomalies are detected, which may reveal damaged or harmed devices. The subsequent development can be visualized as small multiples (see Figure 4) in addition to the monitoring views.

A set of rules maps the input of the field and anomaly detector to color. Saturated and intense yellow, red, and violet represent warnings and alarms. Less saturated colors stand for less serious events, such as gray for normal (uninteresting) status. Some rules provide continues values in addition to ordinal signals. For these rules we use a continuous color scale that varies over saturation and lightness from gray to yellow and over hue from yellow to red. Thus, severe events are perceptually highlighted on the dark background whereas less important events do have less visual impact [22]. The size of elements represents the topological importance of infrastructure elements. We consider central elements (and their dependencies) more important, since their failures are more acute than failures of border elements. Thus, the size of important elements is increased, which also highlights dangers or failures of central elements. We also add domain

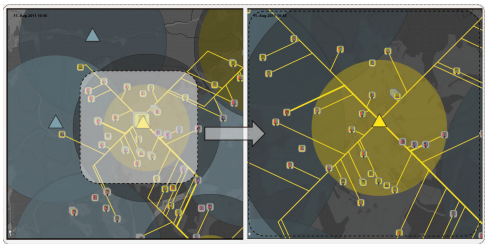


Figure 5. Semantic zooming reveals more details on each zoom level as soon as enough space is available.

details into the symbols such as the current power production and consumption as well as the producer type for transformer stations (see Figure 3(b)).

2) Zoom, Focus & Details on Demand: Two major problems arise when graphs are visualized: the over plotting of nodes and intersection of edges. The over plotting of nodes can be compensated by stacking the overlapping nodes and visualizing them at their average location. They are sorted by their current status. The domain details (e.g., power consumption and producer type) are aggregated and visualized in the foreground element. For the intersection of mobile communications, we omit the painting of connections that are working properly and use edge bundling in order to avoid intersections. These aggregation techniques enhance the readability of the visualization, however, at the cost of hiding or divert some information. The user is therefore able to zoom into areas of interest. If enough space is available, the system will visualize the elements in their normal layout and will provide additional information (see Figure 5).

The user can further interact with the field via control panels, e.g., disabling powerlines or deactivating producers at a transformer station. The user is further enabled to adjust the expected production and consumption for simulation and thus, can create alternatives for decision making.

C. Accessing Human Sensor Information

With the rise of community driven content services, such as Twitter and Facebook, a new information channel for situation awareness has been established in the Web. In contrast to more traditional data sources, like structured sensor data or detailed reports from emergency responders, these new information channels pose novel requirements for data filtering, ranking and aggregation. The relevant information has to be separated from general chatter and organized according to different topic categories. Large amounts of repetitive reports have to be integrated into a consistent and scalable situation overview. In our approach we propose novel methods to address these challenges in order to incorporate social media services as external community driven sensors within the command center environment.

1) Overview and exploration based on automated event identification: The complexity of events and the velocity of streaming data often hinders straightforward situation

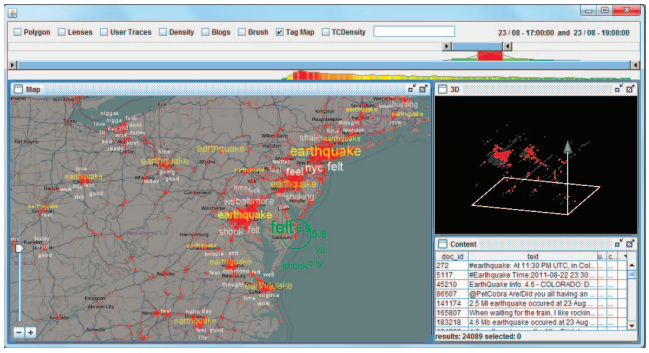


Figure 6. Overview visualization of crisis related topics based on automated anomaly identification. The image shows the social media component with activated anomaly visualization during a large earthquake that happened in August 2011 near Washington DC. The observation of earthquake related events lead to several "earthquake"-clusters in many cities along the coast.

awareness during critical situations. Means for automated detection and display of possibly relevant clues can be a key factor in successfully mastering crisis management. In case of social media data, it is particularly important to detect possible first-hand accounts (e.g. eyewitness information) of on-going situation between large quantities of irrelevant information and to provide visual representations of the discussed topics and observations.

As in [20], we rely on the presumption that messages addressing local events are often of related content and structure and that they are furthermore located in a spatial and temporal neighborhood. This ultimately leads to spatiotemporal clusters of messages reporting on the same situation related topics and keywords. Based on a cluster analysis approach, adapted to the specifics of real-time data, we automatically detect such spatiotemporal anomalies in the continuous data stream. Once a timeframe and geographic region is interactively selected by the analyst, the system generates a map of detected anomalies within that region and timeframe by finding frequent keywords in the message clusters and place them as labels at the corresponding cluster locations on the map. In order to avoid overlapping labels and at the same time show the analyst as much information as possible, we apply a collision avoidance technique that allows overlapping labels to move small distances from their designated locations. Ultimately, the label is not shown on the selected zoom level, if a certain maximum distance for that zoom level has been exceeded.

Our technique provides a broad overview of all events that occur in a given geographic region and, more importantly, an indication of keywords and topics that might be a good starting point for further investigations (see Figure 6). This is particularly helpful if the analyst does not know in advance what to search for or to initially inform him of an unknown ongoing situation. By zooming into the map, our layout technique automatically provides more labels for the given area, as more screen space becomes available for the given

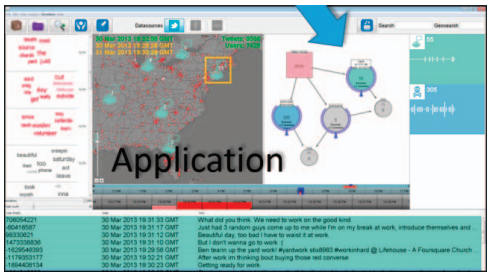


Figure 7. Classification and filtering of crisis related messages. Based on specific information of the past the analyst can load and combine modules from a library of the pre-trained classifiers using set operations. The occurrence of all new messages is shown in real-time. The analyst can associate the modules in the filter combination with specific labels, colors and symbols that are used to highlight messages detected in real time.

region. The analyst thus receives more details of possible sub-events connected to a larger event and can use this as a basis to extend his investigation with traditional textual search, content analysis and focus and context visualizations.

2) Detection of highly relevant information items based on user-steered classification: Besides the need to be informed about unknown or unexpected events, analysts usually also have a distinct domain and area of responsibility and are thus able to define information types that are clearly relevant to their tasks. For example, police officers will always be interested in information about the use of firearms or other acts of violence in their precinct. However, plain keyword-based approaches to find messages fitting to the given information need are often not powerful enough, as the complexity and specifics of language use in social media data can often not be properly reflected.

Especially in real-time analysis scenarios analysts need means to quickly build highly customized filters based on their information needs, their knowledge structure and the specifics of the situation. In our approach we propose a two-step process where a library of Support Vector Machine (SVM) classifiers customized to the specific information needs is trained first, which can then be adjusted and combined with each other and with more simple keyword-, spatial, temporal-, spam- and other filters based on interactive visual set operations (see Figure 7). This idea has already been introduced in [23].

3) Classifier Training: Based on historic data of previous, well understood events, an analyst can explore social media messages to label positive and negative examples for a given event type. This is supported by a range of exploration and analysis tools. Once the analyst has identified a sufficiently large set of example messages related (positive) and unrelated (negative) to the event type, the analyst can label them as such to iteratively progress the semi-automated training process. The training examples are especially useful if they are near the SVM-classifiers decision border, i.e. they have a high probability of being relevant to the topic in terms of keywords, and just the specific combination of terms renders them related or unrelated (e.g. “This morning the power went down.” vs. “I have no power to get out of bed”).

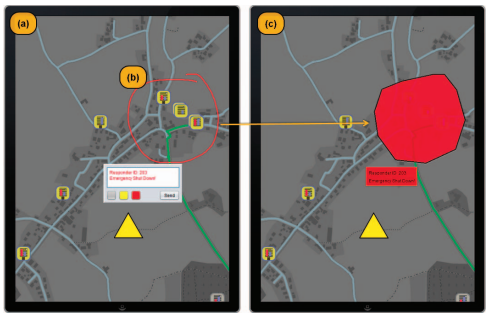


Figure 8. Overview Mobile Concept for First Responders.

4) Real-Time Monitoring: With repeated classifier training analysts can create a comprehensive library of annotated classifier modules for different event and message types relevant to their domain. In a real-time analysis scenario they would usually load and configure a range of classifiers and filters at the beginning of the monitoring period. The most relevant classifiers can be tagged with custom selected colors and symbols in order to highlight corresponding messages if they have been detected. This helps the crisis manager to detect messages related to topics of the current interest in real time, which help to relate human sensor information with events in abstract infrastructures such as smart grids.

D. High Mobility Visualizations & Visual Communication

Mobility for the First Responder is a crucial aspect in their field of work. Therefore, we designed a network visual analytics system that utilizes the advancement of mobile devices (e.g., iPad). Our mobile interface aims to provide an interactive environment where the First Responders would be able to receive detailed information in addition to commands from Site Commanders and Crisis Managers, examine the crisis scenario around them, conduct search and research with clear routine information, and finally provide feedback information to the managers. They have access to the visualizations and information of the command center, which can be focused to their particular location and interest. Section 5 discusses details of our implemented architecture to support these functions.

To help users quickly select and focus on a geospatial region, we developed probing gestures, as shown in Figure 8 (B). This is an extremely important analysis feature because it allows the user to drive the analysis and focus on what is important to their needs on the go. A First Responder can, with their touch enabled glove, directly draw onto the map with his or her finger by drawing a bounding area around a region or mark specific points. The system samples the gestures and computes the convex hull or straight line with linear regression, if demanded. Thus, rapid and noisy drawings are smoothed (see Figure 8(c)). They can further annotate in the selections with real-time updates, as shown

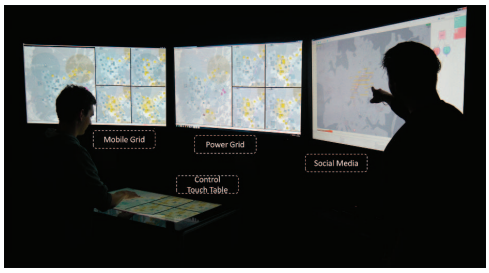


Figure 9. Our control room setup consists of three high resolution displays for the visualizations and a touch table for the steering of clients. At any time it is possible to add further clients.

in Figure 8 (c), and share the information back to their commanders and other responders, through a fast wireless or satellite connection.

V. Control Room

The different components are combined in a control room setup that synchronizes all views and clients on demand and allows the integration of mobile devices.

A. Concepts

Crisis management teams often consist of several experts from different domains. A common way to analyze crisis scenarios is the subsequent analysis of incidents. Typically infrastructures build large graphs and therefore, it is not possible to limit the analysis to a single screen. Hence, we setup a control room (see Figure 9) that supports a distributed and collaborative analysis among several experts. Our setup consists of three high-resolution displays (4xHD resolution per display) and one touch table (Samsung SUR 40). Each display can run and visualize a client that can be steered with the touch table. For example, in Figure 9 the left display shows the power grid and the middle display shows the mobile infrastructure. The user can synchronize the table with each display by pulling their current view/application and perform the application dependent interaction such as changing the viewport or select an infrastructure element. The viewport of the map is then synchronized throughout all clients. Further, the user can change the view or configuration and push the current view on the table to any display.

We see four advantages in this setup: First, single experts use this setup to explore and explain incidents with the aid of different views and visualizations on the crisis scenario as illustrated in Figure 1; these are displayed on the three high resolution displays. Second, the setup can be used to illustrate alternative solutions for decision making: Multiple alternatives and their subsequent development can be visualized simultaneously, which supports experts to draw decisions. Third, if several experts synchronously use this setup, the work can be partitioned on the three displays as well as on the touch table. Every expert receives his own interaction device, for instance a cordless air mouse, which is applied to his own workspace. In case an expert needs to exchange information or enhance visualizations, we offer the possibility to synchronize the clients. Fourth, this setup easily enables a possible combination of the previously named social media component and critical infrastructures.

B. Architecture

In order to enable this vision of a distributed and collaborative environment, there is need for a supportive system architecture. In a collaborative working environment, multiple views and information have to be synchronized. We therefore set up a client-server architecture that supports synchronization across different clients. Our central server manages all connections to the clients and distributes information. Every interaction that needs to be synchronized is first sent to the server. However, the clients do not necessarily need to be synchronized and can also work independently if requested. In addition, the server also handles all connections to the external simulation servers and the local data sources (power grid, mobile grid, weather, and geography). The system requirements include hardware and system independence. In this distributed environment every client running a JAVA VM is able to connect to the server.

Hardware issues. We further support devices that are too weak to render our applications. Depending on their hardware, clients are classified into *complete* or *minimal* clients. *Complete clients* contain enough resources for standalone applications that render the components by themselves. Therefore, the server needs to transfer data and information, used for synchronization, to these clients. *Minimal clients* such as mobile devices do not have enough resources for the whole application and therefore, the server pre-renders the current view of the client according to the device, which is then send and visualized as image. Basic controls are also available such that the image contains the location and type of controls. Thus, *minimal clients* are not updated in real time, however, they have access to the full crisis scenario.

VI. Scenarios

Three scenarios were designed to highlight the need for such systems. Therefore, we designed multiple catastrophes that affect critical infrastructures, such as a mass disease, a cyber attack, as well as a flood-scenario that is presented here. The flood disaster is caused by heavy rain and thunderstorms in a region of Germany. The region is employed with a smart grid by one of the project partners and was flooded in 1987, from which the scenario is inspired.

The scenario starts with heavy rain and thunderstorms. Especially the high grounds of the scenario region are soaked and the soil already begins to become unstable. A thunderstorm in the early morning increased the danger for this area, which alarms the command center and site-commanders of the power grid domain. Due to the social media analysis, which detected messages about unusual water levels, the site-commander sends a power grid technician (first

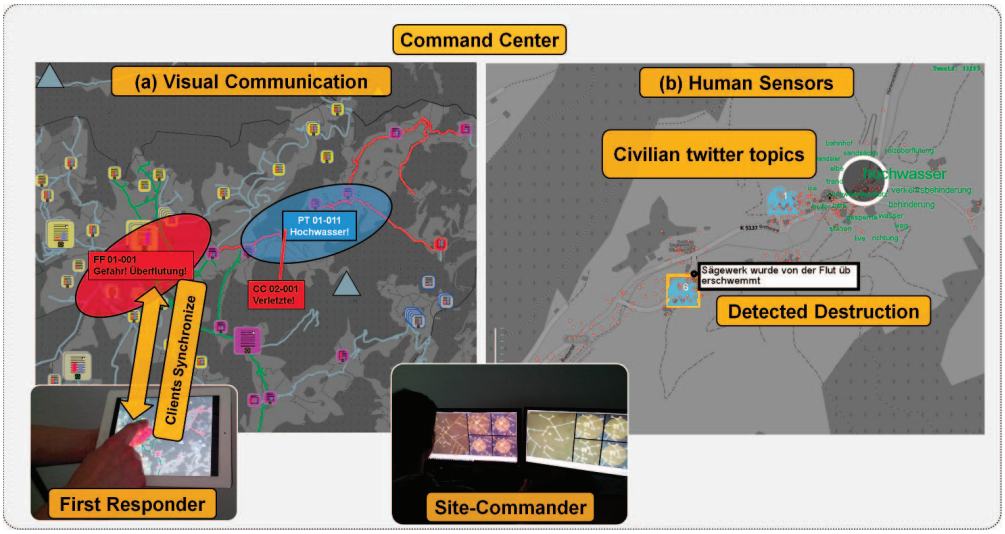


Figure 10. (a) The north-eastern part is flooded, which results in partial blackout (violet stations) and an endangered power grid. First responders update in-situ information about flooded areas and casualties ("Gefahr! Überflutung!", "Verletzte"). (b) The social media analysis early reveals messages about high water levels ("hochwasser") and detects destruction events in real time.

responder) to the region for on-site information. Figure 10 b) shows the detected messages of amblers about high water levels ("hochwasser"). Also, the command center reacts with alarming the regional response teams; in this case regional fire-fighters that are equipped with *mobile analysis devices*.

A debris avalanche hits a small village in the suburban region. Many homes are flooded and separated from the center of the region. The debris avalanche also hits a bridge and the flood is jammed. Transformer stations in the eastern parts are immediately destroyed and the blackout cascades from the east to the south (violet stations). The remaining power circuit in the west and north is suffering from this immediate loss of consumption, which raises the voltage levels (see yellow stations in Figure 10). The fire-fighters cannot reach the casualties since streets and fields are not passable. They update the flood-endangered areas and the status of streets and casualties (see Figure 10 (a): "Gefahr! Überflutung!", "Verletzte"). They request the context information, which is enriched by the technician who supports additional information from his location. Both teams update the information, which is synchronized between their clients and the central control room. The endangered areas and evacuation routes are coordinated by the command center and send to site-commanders and first responders. The fire-fighters begin to evacuate people to the south. The water level still raises and reaches over the bridge. The endangered area where one team of fire-fighters is located is flooded. The region is now suffering under a blackout since the central transformer station is hit. Evacuation routes and endangered areas are dynamically updated by first responders and command center.

After the situation stabilizes, the power grid site-commander and the technician coordinate how to ensure that most of the region can be power supplied. Therefore, the technician updates the status of transformer stations and informs the site-commander, which station can be switched on. Further, the repair and response teams need to organize, which streets and transformer stations have to be repaired. Updating the on-site information to the command center and site-commanders enables their situational awareness and allows directing forces where they are needed. They can consider local incidents in their decision making, which is effective with our system.

VII. Qualitative Evaluation

Field studies for crisis management systems are hard to conduct. Realistic crisis data is often not available or classified. Therefore, the project partners decided to simulate realistic scenarios, which are then analyzed by target users with our system. We conducted a qualitative study based on expert feedback rounds and interviews.

A. Process

Evaluation Teams. We formed four teams within the nationwide interdisciplinary project: 1) *Data team* consists of two members of the Federal Office of Civil Protection and Disaster Assistance of Germany (henceforth, BBK), as well as four representatives of power suppliers, and further two simulation experts for smart grid technology and social media. They designed the scenarios mentioned above and provided the data. 2) The visual *design team* (represented by the authors) consists of eight visual analytics experts who designed the system based on the scenarios and data. 3) An *interview team* of two persons with backgrounds in visual design conducted the qualitative interviews with domain experts. 4) An external *experts team* consisting of two members of the BBK with experience in crisis response and ten power grid operators of different regional power grids

in Germany. We selected these different but related domains since they can be considered as the future target users. The crisis managers are part of the command center level, whereas the operators can be considered as site-commanders that are focusing on one particular affected region and domain.

Interviews. The *interview team* was involved in the creation of the scenarios but not in the design process and is therefore considered independent of the design decisions. However, this team was trained by the *design team* on the system components and supported with documentations. Further, they prepared questionnaires for the qualitative interviews. The *interview team* visited seven control rooms of regional power grid suppliers in Germany and also interviewed experienced crisis managers of the BBK. This *expert team* did not know the system and scenarios. The interviews were conducted in concrete steps: First, the *interview team* presented the single components of the system. After the experts were familiar with the system the interview team presented one scenario as a use case. The scenario was stopped at critical events and the experts were asked to analyze the crisis and to draw decisions with the help of the system. Then interviews according to the questionnaires were conducted. The *interview team* analyzed the results and summarized the findings, which were reported back to the *design team*, who carefully analyzed the findings and improved the system for a second iteration (future work).

B. Results & Discussion

1) Domain experts reported that such systems are needed for crisis response. The experts reported that there is an urge for information of the crisis site, because in most cases the command centers are blind and wait for phone calls: “The social media analysis is great. In most cases, first responders are too busy to update the crisis center. Direct access to Twitter messages that are linked to the crisis would give us a clearer and faster overview of the crisis.”, “We want a direct visual communication with first responders”. We found that they are not only interested in the information of first responders about the crisis but also how the affected civilians are describing their status: “Some people may just feel to be forgotten by the government and we could respond with sending in some teams to show our presence”. Further, we found that first responders do have an urge for the context and development of the crisis, because they do not know what may hit them within the next minutes. They highlighted that in-situ and social media analysis can improve to narrow down root causes on-site and also to effectively steer first responders:“The control center does not even know, which transformer station can be reached by repair teams. We would need information about streets and areas around stations. If we could use these tools today, we could directly send the teams where they really could make a difference”. The *interview team* found that the expert team efficiently understood the tools, however, they highlighted that a target user of such systems would require a significant amount of training. They conclude that the concepts of our system are sound, however, will require further investigations to integrate this in future crisis response centers.

2) Crisis-Managers and Site-Commanders disagreed on the level of detail. All experts agreed that social media and linked communication with first responders is important and that our system could be used in crisis scenarios. We found that power grid operators and crisis managers disagreed on the level of details of such visualizations. The crisis managers wanted to perceive the crisis and simulate different alternatives in abstract manner. Thus, they were satisfied with our components. The power grid operators requested more domain details and domain standards in the overviews. They reported that the whole system is interesting; however, the visualizations do not meet the requirements of power grid monitoring. We conclude that our architecture must provide links to established domain systems. These interfaces must provide abstractions of domain details for the communication between the domain dependent site-commander and the crisis managers in order to adapt the level of detail.

C. Discussion, Limitations & Future Work

We found that our target users were convinced of the system and its applicability. However, we see that the system does not fulfill all requirements to be an operational system for power grid control. Research has evolved over decades to develop customized solutions for this particular infrastructure. Interestingly, the operators that were involved in the flood disaster that hit Germany in 2013 said that they were almost blind after the water destroyed the first transformer stations. Therefore, they highlighted an urge for on-site information and social media analysis. In the future scenario of interconnected infrastructures such as smart grid technology, we see a higher complexity as in today’s power grids. Command centers must overlook and perceive the full context of a crisis. Therefore, abstract visualizations are needed, which was approved by our crisis managers. We argue that our system exemplifies a means for future central crisis managements to integrate different critical infrastructures, social media and in-situ analysis. It will be interesting to discover the correct level of detail to satisfy each role in the command structure. For this, we plan to conduct user assessments to improve our components to the needs of particular site-commanders. In addition, we will focus on interfaces to established domain solutions and to develop means for a seamless communication between the levels. Another issue is security. The architecture might be vulnerable although we encrypt the communication between clients and server. Further, the issue of in-feeding wrong information with, e.g., a stolen device or misleading Twitter messages was raised in our interviews. Therefore, we see an urge to include security protocols into this architecture.

VIII. Conclusions

In this paper, we present a visual analytics system that combines multiple critical infrastructures, social media and in-situ analysis to support the different levels of command structure in crisis response. We present specialized equipment and visualizations for control rooms and mobile devices. We discuss means for interactive simulation and visualization of the development of a crisis. This enables interdisciplinary and distributed teams to understand and respond to crisis situations. Our system was applied in realistic scenarios and presented to crisis managers, who conclude that there is an urge for such systems for crisis response.

References

[1] "VASA," 2011 – 2014, funded by the German Federal Ministry of Education and Research (BMBF) under the grant Visual Analytics for Security Applications.

[2] "VASA," funded by the U.S. Department of Homeland Security's VACCINE Center.

[3] P. Proulx, S. Tandon, A. Bodnar, D. Schroh, R. Harper, and W. Wright, "Avian flu case study with nspace and geotime," in *IEEE Symposium on Visual Analytics Science And Technology*. IEEE, 2006, pp. 27–34.

[4] J. Kohlhammer, T. May, and M. Hoffmann, "Visual analytics for the strategic decision making process," in *GeoSpatial Visual Analytics*. Springer, 2009, pp. 299–310.

[5] R. Maciejewski, S. Rudolph, R. Hafen, A. Abusalah, M. Yakout, M. Ouzzani, W. S. Cleveland, S. J. Grannis, M. Wade, and D. S. Ebert, "Understanding syndromic hotspots-a visual analytics approach," in *IEEE Symposium on Visual Analytics Science and Technology.* IEEE, 2008, pp. 35–42.

[6] V. Pascucci, D. E. Laney, R. Frank, G. Scorzelli, L. Linsen, B. Hamann, and F. Gygi, "Real-time monitoring of large scientific simulations," in *Proceedings of the 18-th annual ACM Symposium on Applied Computing*, Melbourne, Florida, March 2003, pp. 194–198.

[7] E. Santos, J. Freire, C. Silva, A. Khan, J. Tierny, B. Grimm, L. Lins, V. Pascucci, S. A. Klasky", R. D. Barreto, and N. Podhorszki, "Enabling advanced visualization tools in a simulation monitoring system," in *Proceedings of the 5th IEEE International Conference on e-Science*. IEEE, December 2009, pp. 358–365.

[8] B. Campbell and C. Weaver, "Rimsim response hospital evacuation: Improving situation awareness and insight through serious games play and analysis." *IJISCRAM*, no. 3, pp. 1–15.

[9] S. Kim, Y. Jang, A. Mellema, D. Ebert, and T. Collins, "Visual analytics on mobile devices for emergency response," in *IEEE Symposium on Visual Analytics Science and Technology.*, Oct 2007, pp. 35–42.

[10] T. J. Overbye and J. D. Weber, "New methods for the visualization of electric power system information," in *IEEE Symposium on Information Visualization*. IEEE, 2000, pp. 131–16c.

[11] P. C. Wong, K. Schneider, P. Mackey, H. Foote, G. Chin, R. Guttromson, and J. Thomas, "A novel visualization technique for electric power grid analytics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 3, pp. 410–423, 2009.

[12] F. Fischer, J. Fuchs, P.-A. Vervier, F. Mansmann, and O. Thonnard, "Vistracer: a visual analytics tool to investigate routing anomalies in traceroutes," in *Proceedings of the Ninth International Symposium on Visualization for Cyber Security*. ACM, 2012, pp. 80–87.

[13] J. Liu, Y. Xiao, S. Li, W. Liang, and C. L. Chen, "Cyber security and privacy issues in smart grids," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 4, pp. 981–997, 2012.

[14] J. J. Rooney and L. N. V. Heuvel, "Root cause analysis for beginners," *Quality progress*, vol. 37, no. 7, pp. 45–56, 2004.

[15] A. B. Marques, G. N. Taranto, and D. M. Falco, "A knowledge-based system for supervision and control of regional voltage profile and security," *IEEE Transactions on Power Systems*, vol. 20, no. 1, pp. 400–407, 2005.

[16] D. Wigdor, H. Jiang, C. Forlines, M. Borkin, and C. Shen, "WeSpace: the design development and deployment of a walk-up and share multi-surface visual collaboration system," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2009, pp. 1237–1246.

[17] H.-C. Jetter, M. Zöllner, J. Gerken, and H. Reiterer, "Design and implementation of post-WIMP distributed user interfaces with ZOIL," *International Journal of Human-Computer Interaction*, vol. 28, no. 11, pp. 737–747, 2012.

[18] B. M. Tomaszewski, A. C. Robinson, C. Weaver, M. Stryker, and A. M. MacEachren, "Geovisual analytics and crisis management," in *Proceedings of the 4th International ISCRAM Conference*. Delft, the Netherlands, 2007, pp. 173–179.

[19] A. M. MacEachren, A. Jaiswal, A. C. Robinson, S. Pezanowski, A. Savelyev, P. Mitra, X. Zhang, and J. Blanford, "Senseplace2: Geotwitter analytics support for situational awareness," in *IEEE Conference on Visual Analytics Science and Technology*. IEEE, 2011, pp. 181–190.

[20] D. Thom, H. Bosch, S. Koch, M. Worner, and T. Ertl, "Spatiotemporal anomaly detection through visual analysis of geolocated twitter messages," in *IEEE Pacific Visualization Symposium*. IEEE, 2012, pp. 41–48.

[21] S. Mittelstaedt, D. Spretke, D. Sacha, D. A. Keim, B. Heyder, and J. Kopp, "Visual analytics for critical infrastructures," in *Proceedings of International ETG-Congress 2013; Symposium 1: Security in Critical Infrastructures Today*. VDE, 2013, pp. 1–8.

[22] L. Wang, J. Giesen, K. T. McDonnell, P. Zolliker, and K. Mueller, "Color design for illustrative visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1739–1754, 2008.

[23] H. Bosch, D. Thom, F. Heimerl, E. Puttmann, S. Koch, R. Kruger, M. Worner, and T. Ertl, "ScatterBlogs2: real-time monitoring of microblog messages through user-guided filtering," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2022–2031, 2013.

Anomaly Exploration and Visual Analytics of Financial Data

Shehzad Afzal* Purdue University
Isaac Cho† University of North Carolina
Calvin Yau* Purdue University
Junghoon Chae* Purdue University
Sungahn Ko* Purdue University
Abish Malik* Purdue University
Kaethe Beck* Purdue University
William Ribarsky† University of North Carolina
David S. Ebert* Purdue University

Abstract—Finding anomalies in complex financial data is a challenging task. Identification of anomalous behavior in financial data requires analyzing group behavior in stock trading and identifying those trades that deviate from group behavior. Once anomalous patterns are identified, an analyst needs to review any contextual evidence available through news media and other relevant sources to understand whether these anomalous patterns are explained or not. Conventional charts (e.g., line graphs) traditionally used in financial data analysis provide limited support to detect and explore such financial anomalies, especially in the absence of any related contextual information. In this work, we have collaborated with groups of financial analysts and identified tasks, goals and requirements of a visual analytics system that could be used to detect and explore anomalies in financial data. Our system integrates state-of-the-art visualization techniques combined with automated anomaly detection, enabling an anomaly-driven exploration of financial data. We demonstrate the effectiveness of our system with case studies and feedback from groups of domain experts.

1 Introduction

Determining and understanding financial anomalies is a challenging task for financial analysts due to the size and complexity of the market data. Financial anomalies refer to empirical results that appear to be inconsistent with maintained notions of efficient markets and those of standard asset-pricing model behaviors [38]. Examples of financial anomalies include abnormal behaviors in stock trading and deviations in the pricing of securities compared to their corresponding sectors, indicating inconsistent patterns. Anomalies can be varied and can be caused by a multitude of factors; some may appear to occur sporadically, while others may exhibit trends or periodicity (e.g., [7, 9]). For example, abnormal stock pricing behaviors can occur due to faulty product lines, rumors, changes in supply and demand, corporate management issues, financial attacks, or insider trading as well as many other contributing factors. The long term impacts associated with the different types of anomalies can also be varied. This makes it critical for analysts to identify and mitigate the effect of these aberrant behaviors. There is a need to provide analysts with the tools necessary to probe these anomalies and to identify, explore, and understand their causes and behaviors.

To address these challenges, we present a visual analytics approach [42] for exploring and analyzing financial anomalies. Our system has been designed using a user-centered approach [1] to address the needs of financial analysts responsible for monitoring abnormal financial market behaviors. Further, we conducted an extensive survey of financial analysis and anomaly detection literature to identify gaps in the current practices during our design process. Our visual analytics system provides several coordinated views to enable the exploration of financial anomalies. We link the anomalies with contextual information (e.g., news stories) to allow analysts to explore their potential cause and behaviors. The major contributions of this paper are as follows:

- Design requirements from financial analysts for anomaly triggered exploration of financial data
- Qualitative feedback on the design requirements and the system from domain experts
- An anomaly based visual analysis approach that provides contextual information to explain financial anomalies

*e-mail: {safzal|yauc|jchae|ko|amalik|kaethe|ebertd}@purdue.edu
†e-mail: {icho1|ribarsky}@uncc.edu

Manuscript received 31 Mar. 2014; accepted 1 Aug. 2014; date of publication xx xxx 2014; date of current version xx xxx 2014.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

The remainder of the paper has been organized as follows: Section 2 provides related work on financial analysis. Section 3 provides the details of the problem domain. Section 4 provides the design requirements of the problem domain. Section 5 describes our anomaly based analysis approach. Section 6 introduces our visual analytics system, the functionality and reasoning of each component and how they tie in to the design requirements. Section 7 provides two use cases of our system in the oil and technology sectors. Section 8 provides a discussion on our design process and the feedback from domain experts. Finally, Section 9 presents conclusions and future direction.

2 Related Work

As industry professionals have expressed [39], financial data is typically time-series data. In analyzing the time-series data, standard charts (e.g., line graphs and bar charts) have been commonly utilized. A well-known time-series data analysis method is *"technical analysis"* [13] where straightforward graphical charts are used to display financial indicators over time [31]. The main disadvantage of the standard charts is scalability. When there are many lines, the visualization is quickly cluttered.

2.1 Visualization Techniques for Financial Data

In order to analyze different types of financial data, various visualization techniques have been utilized. Scatterplot and scatterplot matrices [15] have been used mainly for stock analysis. A popular approach in using the scatterplot is to change color, size, shape, and orientation of the points [41, 35, 46] presenting individual items in financial data.

Dense (or pixel-oriented) displays map a value to a color-coded pixel in order to best utilize the screen space. Ziegler et al. [48, 49] computed performance and growth of assets (bond and fund) for all possible time intervals and presented them in a single dense view. Their matrices are so dense that their visualizations enable users to compare more than 12,000 assets simultaneously. Pixel Bar Charts [25] are a technique that transforms a line graph into a bar filled with densely color-coded vertical lines. A series of these charts is utilized to present metrics of market volatility [47]. MarketAnalyzer [27] uses relatively large pixels in multiple coordinate views, allowing competitive advantage analysis based on sales data. Both WireVis [8] and the work of Alsakran et al. [3] used larger pixels for wire transactions and mutual fund [2] analysis but used different ratios when width and height of pixels in visualizations changed.

In Treemaps, a dimension (e.g., a market sector) is connected to a rectangle whose properties (e.g., size, color, label and position) are determined by the connected dimension. This approach is proven to have advantages for categorization and rank [29], but also has weaknesses in providing developments of markets or assets over a long time period, and presenting information of small assets or companies. The most popular use of treemaps is for stock market analysis [44], where the volume, capital size or risk of each sector determines the size of each rectangle.

An early tutorial survey on business information visualization was presented by Tegarden [40] with a focus on industry's visualization systems. In order to evaluate which visualization techniques (classic charts, parallel coordinate plot, icon display, recursive pattern, and treemaps) are more suitable to which tasks, Merino et al. [29] performed empirical performance experiments, where classic charts are proven to be advantageous in visualizing stock market. We, therefore, extensively utilize line charts in this work for presenting stock market data with anomalies. More recently Dang et al. [12] proposed a functional framework for evaluating financial visualization products and Flood et al. [16] discussed potential benefits of visual analytics [42] in monitoring systemic financial stability. In contrast, Lemieux et al. [28] describe visual analytics components combined with financial risk analysis. But, the anomaly detection approach has not been utilized in the past work. Lastly, Dumas et al. provide an excellent online survey[1] for financial visualization systems.

2.2 Data Mining Techniques for Financial Data

The increasing size and complexity of the financial data have caused analysts to spend more time on finding suspicious points in the data and collecting evidence to validate their hypotheses. In order to help the analysts, various techniques in data mining and statistics have been developed for detecting anomalies. Because anomaly detection on time-series is a very broad topic and beyond the scope of this work, we only describe the visualizations combined with anomaly detection algorithms. There are good surveys for anomaly detection algorithms on time-series data including [10, 20].

One common approach to find anomalies is to use clustering algorithms (e.g., k-means [47] and Self-Organizing Map [37, 36]) that group data items based on algorithms' criteria (e.g., similarity). This approach is effective when a result of the algorithms is much smaller than the original data because finding features and trends in the groups can require less time compared to the task of investigating individual items in data. Dimensional reduction algorithms including multidimensional scaling (MDS) also have been utilized in order to transform high-dimensional data into low-dimensional data. Scatterplots are a good candidate for using this transformed data for visualizations (e.g., with MDS [6, 19]). In this work, since we have natural clusters of the information in terms of sectors of individual stocks, we do comparison of individual stocks to sectors as an anomaly indicator.

Predictive analytics[2] based on statistical algorithms is gaining popularity for analysis and prediction of various types of data (e.g., movie rating, a death number from diseases, disease spread, and vehicle traffic). For financial data analysis, specific models and theories are often developed in order to expose special features or trends in data including Performance/Weight Matrix [49], portfolio theory [32], and economic decision-making models [34]. Although clustering and statistical algorithms have helped analysts performing tasks of different data sets, they still need a visual tool where they can fully take advantage of synergy produced by combining anomaly detection algorithms and visualizations. In this work, we present an efficient analytical environment where automatically detected anomalies are presented in visualizations in conjunction with auxiliary information in order to reduce analysts' burden generated in anomaly detection and evidence validation processes.

[1]FinanceVisBrowser, http://financevis.net/, (accessed March 25, 2015)

[2]IEEE VIS 2014 Visualization for Predictive Analytics, http://predictive-workshop.github.io/, (accessed March 25, 2015)

3 Data, User, Requirement, and Task Analysis

Anomaly-driven data analysis is a commonly utilized approach in visual analytics where interactive visualizations are combined with analytical algorithms and the human in the analysis loop drives the process. The goal of this approach is to create synergies between the users and the computational methods that identify anomalies (e.g., abnormal event detection in text data streams [43]). An example of this approach is a visual exploration model [26] where visual interfaces allow users to change parameters of data models and visualizations, generate new visualizations, and produce new insights.

In designing our system for anomaly-driven data analysis, we used an iterative, user-driven, design approach. As an initial step, we held two overall work flow and process design sessions with analysts (task-level knowledge elicitation), followed by system prototyping with mock-up design generation. Next, we held user sessions for requirement definition using our system mock-ups and cut outs of the different visual components.

We held three sessions to obtain system requirements with three focus groups of financial analysts, with focus group size ranging from 2 to 9 analysts. The analysts had diverse backgrounds from economics and social science, to quantitative analytics and statistics. Their current roles also varied including many who were interested in predictive analytics.

Our first focus group consisted of analysts who monitor financial datasets in order to detect abnormal situations (e.g., unusual variations in stock prices), and to generate reports with evidence for further action. Analyzing such abnormal situations is itself a complex and time consuming task. It requires analysts to look into contextual evidence available through news data sources to see if there is any causal relationship between an event or political situation and the anomaly.

Specifically, these analysts were interested in extracting the following information from the identified financial data anomalies:

R1 A context to the anomalies provided from relevant data sources (e.g., news stories).

R2 Information about the contextual evidence of the anomalies and whether the identified fluctuations are statistically significant.

R3 Historical patterns of the financial stock market data (e.g., by day, week, month, year).

R4 Information about where the change came from (e.g., which companies generate anomalies?).

R5 Thresholds set in order to drill down to anomalies whose ranks are significant.

R6 The ability to distinguish regular/periodic properties in data from anomalies.

Our second focus group consisted of analysts responsible for generating reports on future financial trends.While the initial procedure of detecting anomalies is similar to that of the first focus group, this group requires interactive comparisons of the different signals and the obtained anomalies. The additional requirements that we identified from this focus group included the following:

R7 Comparison of stock market data and anomalies across countries, other stocks, and corresponding sectors.

R8 The rarity of anomalies at different aggregation levels (e.g., how many anomalies did a company generate?).

Finally, we interviewed an economist responsible for analyzing large scale financial trends (e.g., sector-, nation-wide trends). The last requirements were similar to those of the previous groups; however, the analyst stressed the importance of providing contextual evidence about anomalies that helps an analyst confirm certain hypothesis (R2,R5). This analyst was also interested in comparing the anomalies across different aggregation levels (e.g., individual ticker vs. corresponding sector) (R7,R8), and support for users seeking explanations regarding anomalies (R1).

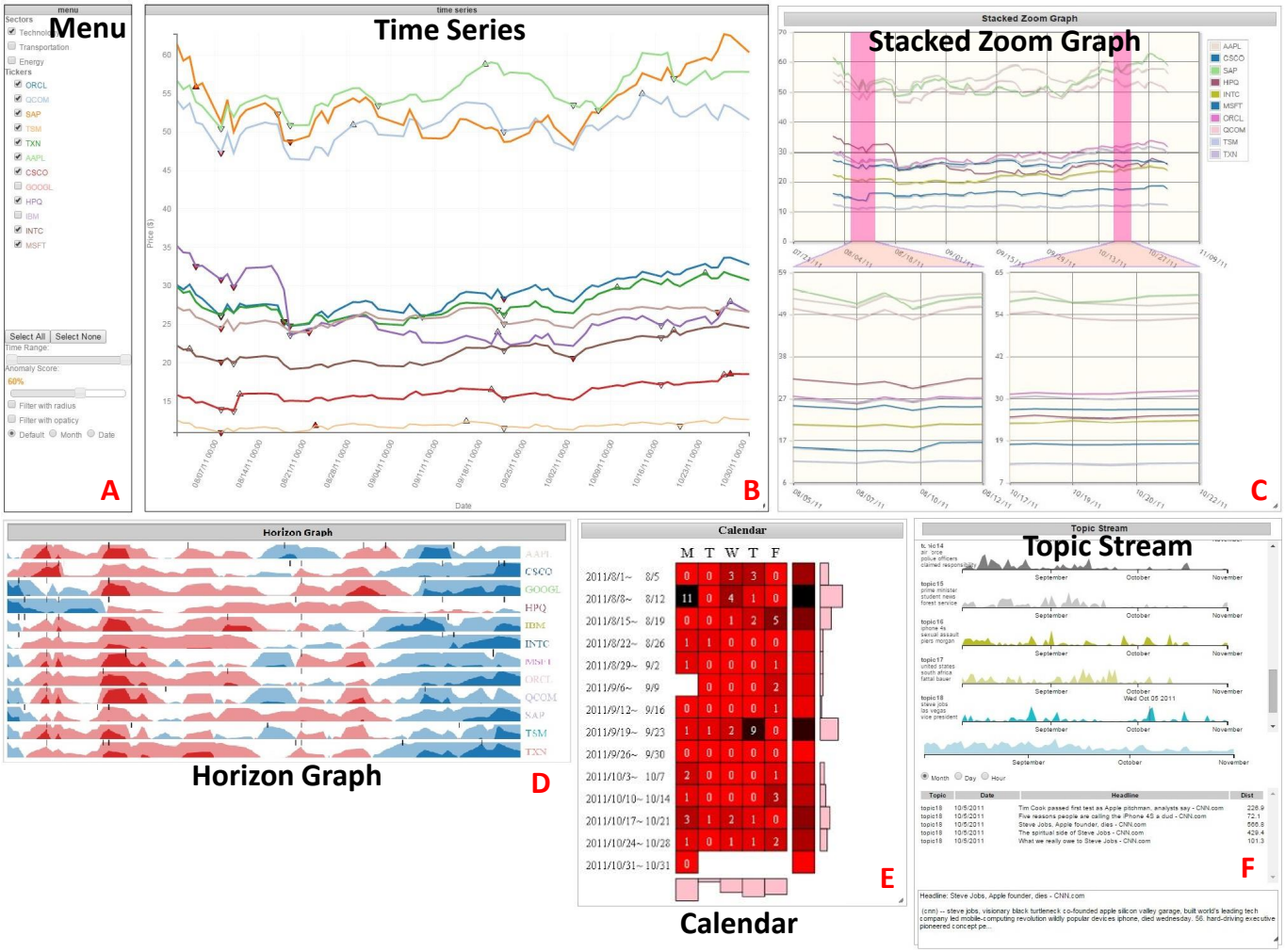


Fig. 1. Overview of our system consisting of Menu (A), Time series (B), Stacked zoom graph (C), Horizon graph (D), Calendar (E), and Topic stream (F). All components are tightly linked each other.

Based on our discussions with these analysts, we designed the visual analytics system described in this paper to support their tasks for anomaly-driven financial data exploration and analysis. Our visual analytics system provides users with important context information of the detected anomalies which helps them understand their significance. It also provides historical patterns of the anomalies in different market sectors, and a context for the anomalies using news sources. The resulting system provides a unified visual analytics environment where analysts can quickly find and extract necessary information from unstructured, multisource data using an overview and details on demand approach.It also enables them to connect anomalies to relevant patterns, as well as generate reports with contextual evidence.

4 Visual Analytics Financial Assessment Environment

Our visual analytics environment consists of a series of linked views, and has been built to operate for both historical and real time financial market datasets. The system has been implemented in HTML, PHP, and Javascript using D3[3], Protovis[4]and jQuery UI[5].Our system has been organized into a traditional dockable dashboard view: each window can be closed, moved, and resized as desired. Figure1 presents a snapshot of our system. The main components include the following:

- **Time Series View**: Presents ticker prices and anomalies over time and provides users with display and configuration options
- **Horizon View**: Presents ticker prices, anomalies, trends for individual stocks, market/sector indices, and the overview trends
- **Calendar View**: Presents count of anomalies in the format of a calendar in order to explore seasonal and cyclical trends
- **Stacked Zoom View**: Presents ticker prices over time in detail and its relation to the overview
- **Topic Stream View**: Presents related news stories for each topic/ticker over time

Additionally, the system provides a control panel that allows users to interactively select their stock market tickers of interest. This facilitates the analysts' ability to display the ticker data in the time series visualizations, as well as the other components of the system. The control panel also provide users with the ability to select the time range and temporal aggregation levels (e.g., hour, day, week, month). The sub-sections that follow provide details of the different views in our visual analytics environment.

For the news story data, we collected news stories from the CNN News network by our customized webpage crawler that extracts news text from their web domain. Our crawler system has been built using

[3]http://d3js.org/,(accessed March 25, 2015)
[4]http://mbostock.github.io/protovis/,(accessed March 25, 2015)
[5]https://jquery.com/,(accessed March 25, 2015)

Python and Scrapy[6]. For each news story, we extract titles, story text, and time information of articles. The data is then stored into an Apache Solr server[7] for post processing.

4.1 Time Series View

The main view of the system consists of the time series view that initiates the analysis process and provides a temporal overview of the financial data and anomalies to the analysts. The time series visualization displays values and variations in stock prices over different temporal aggregation levels (R3), along with anomalies detected for each ticker using the methodology described in Section 5. The time series display supports interactive brushing and linking with other linked views. Hovering the mouse over a specific time series highlights the ticker and provides further details on demand (e.g., ticker name, stock price at that time step).

We apply a glyph-based approach to display anomalies identified for each ticker using the method described in Section 5 (R2). A confidence score for each anomaly is calculated from the time-series of market prices based on the z-score value as explained in Section 5.1. It should be noted that we allow users to interactively adjust threshold for this z-score value in the system (R5). In order to display anomalies on the time series displays, triangles are drawn at time steps where anomalies are detected. We use triangles that are pointing up to represent anomalies that indicate abnormal positive growths in the signal. Similarly, we use triangles that are pointing down to represent anomalies with an abnormal negative fall. The anomalies that are above user provided threshold are highlighted with a solid red colored triangle. Users are also provided with different visual encoding options to display anomalies. For example, users may choose to encode the z-score values for the anomalies using the glyph size and opacity. Users can also choose to hide certain investigated anomalies in order to focus their attention on anomalies (as shown in (B) and (C) of Figure 2) with low confidence scores. Finally, we note that users can click on the individual triangles to select a specific anomaly for further investigation in the other linked views.

4.2 Horizon Graph View

Line graphs allow users to observe trends in temporal data. However, as the number of ticker signals increase, it becomes difficult for users to compare the trends across the different signals. In order to alleviate these challenges and enable simultaneous comparisons of stock prices and anomalous regions (R7), we adapt a two-tone pseudo coloring technique in the horizon graph [33]. Our implementation of the horizon graph is shown in Figure 1 (D). Here, the different stock time-series are vertically arranged. Each time series signal is then split into two color tones by flipping, using the average line of the time-series. Here, the upper half (i.e., values above average) are colored in blue, while the lower half (i.e., values below average) are colored in red [33]. The height of the time series is reduced further by subdividing each half into two bands. The number of bands used for each half are configurable by the user, and it also depends on how many stocks we need to analyze simultaneously. Interested readers can find more details on the implementation of the horizon graph[8]. Besides displaying the stock prices, we have extended this view to visualize anomalies by drawing vertical bars on top of each ticker graph as shown Figure 2 (E). We note that overlaying anomalies atop each horizon graph enables users to visually detect correlations between anomalies and extremes in time series values (R6). This view helps analyst identify sector trends in both anomaly and ticker data. Analyst could easily identify anomalous regions and could explore further details of such regions by utilizing the stacked zoom graph. The horizon graph acts like a summary view helping users to identify any patterns in sector stock prices and anomalies.

[6]http://scrapy.org/,(accessed March 25, 2015)

[7]http://lucene.apache.org/solr/,(accessed March 25, 2015)

[8]http://www.stonesc.com/Vis08_Workshop/DVD/Reijner_submission.pdf,(accessed March 25, 2015)

4.3 Calendar View

The calendar representation, as shown in Figure 1 (E), is intended to show the relationship between the number of anomalies over the user selected calendar year for the selected ticker indices. Here, the total number of anomalies observed across dates are displayed in the format of a calendar [45] where each row corresponds to a week, and each column corresponds to the day-of-the-week. Note that our calendar view consists of only 5 business days (Monday through Friday) because the stock markets are closed over the weekends. We apply a sequential color scale [21] to encode the total number of anomalies per day. The bar charts at the end of each row and column encode the total number of anomalies for each week and day-of-the-week, respectively. The calendar view reveals seasonality and cyclic trends, and enables users to explore the anomaly patterns by days-of-the-week and across different weeks (R3).

4.4 Stacked Zooming View

In the analysis of temporal data, one of the challenges is exploring multiple disconnected time frames in order to discern temporal patterns. Financial analysts are frequently required to examine temporal trends in their data, and also to determine whether any recurring patterns occur in the data (R3, R6, R7). To support these analytic tasks, we provide analysts with a stacked zooming line graph visualization technique [24]. This technique supports multi-focus interactions that allows users to visualize multiple parts of time series signals by juxtaposing them at subsequent levels [14]. In order to preserve context among the different levels, we implement a hierarchical approach that provides visual cues to connect the parent graphs to corresponding children graphs. This approach enables users to quickly navigate and find corresponding child graphs for their selections.

4.5 Topic Stream View

The topic stream view allows users to explore through news stories related to ticker events and time of interest [Leadline]. We extract meaningful topics from text corpora by LDA [5], which is widely used for topic modeling. The topic stream view presents the volume of news stories for given topics for a time range selected by the user in the top section. By clicking on a specific time of a specific ticker, or by directly selecting an anomaly in the time series view, the topic stream view will present associated topic streams. Then the user can find topics and news stories corresponding to the same time range on the ticker. When a news story of interest is selected, the complete story along with its related topics will be displayed in the bottom section. By perusing the news stories, the user can make sense of them and their topics, connect them to particular anomalies, collect relevant stories for a more complete picture, and hypothesize a detailed explanation for the detected anomaly. This visualization supports two modes of grouping and listing topics. First mode is based on LDA topic modeling and second mode is based on company-based filtering. In company-based filtering, news stories are filtered based on company specific keywords. The topic stream view then presents the volume of news stories for given companies within user selected time range. All news articles are grouped based on the mode selected by the user.

5 Anomaly Analysis

In this section, we discuss our anomaly analysis techniques for detecting abnormal financial market behaviors. A large number of studies have documented that the distribution of stock returns shows a day-of-the-week effect [17, 18]. We propose three different automated techniques which take account of the presence of seasonality in stock returns. In our system these automated techniques are effectively coupled with our visual analytical environment. Analysts are able to explore the contextual information surrounding such available anomalies. Our system allows the analysts to choose one of the techniques based on their analytical purposes and personal preference.

5.1 Seasonal Trend Decomposition

The existence of seasonal variations in production and sales is well known [22]. Similarly, the stock market also exhibits seasonal pat-

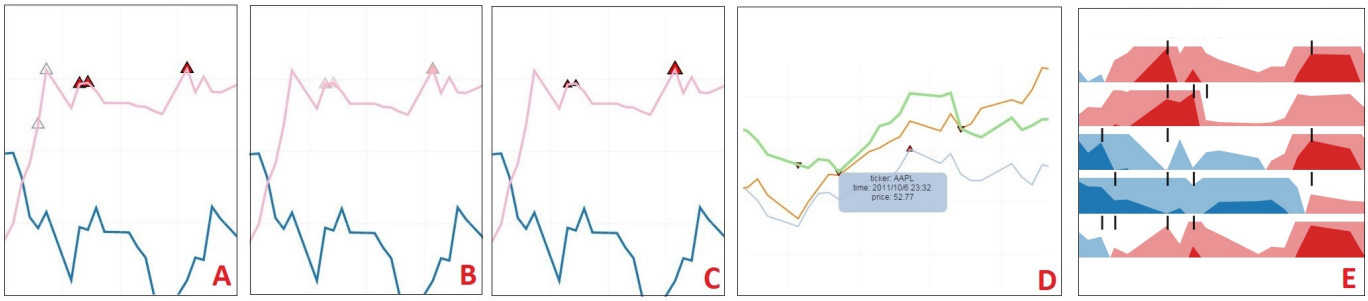


Fig. 2. Glyph based approach for encoding anomaly scores. Different ways of encoding z-score. Empty or filled triangles (A), Transparent or opaque triangles (B), Large or small triangles, (C), Tooltip showing ticker details (D), Horizon Graph: Black bars represent anomalies (E).

terns and trends at certain times of the day, week, and month [4]. In order to identify abnormal stock behaviors that can be indicative of unusual behaviors, we utilize the Seasonal-Trend decomposition based on Locally-weighted regression (STL) approach [11]. The STL technique decomposes the time series into three components: a trend component, a seasonal component, and a remainder, as shown below:

$$Y = T + S + R \tag{1}$$

Here, Y is the original time series of interest, T is the trend component, S is the seasonal component, and R is the remainder component. We utilize this approach to model the financial time series in order to account for the various components, and also to detect irregular components in the signals. The resulting estimates of the trend and seasonal components are then used to compute the remainder: $R = Y - T - S$. In our approach, we utilize large deviations from the model behavior (determined using large remainder R values) to detect substantial variations in the time series. We define these deviations as anomalies. To calculate these deviations, we utilize a seven day moving average of the remainder values to calculate z-scores in order to locate peaks and outliers. The z-scores for each ticker are given by $z = (R(d) - \sigma)/\mu$, where $R(d)$ is the remainder value for day d, σ is the mean remainder value for the last seven days, and μ is the standard deviation of the remainders. Finally, if the z-score values are higher than 3 for day d, we consider the value to be abnormal within a 99% confidence interval.

5.2 Exponentially Weighted Moving Average

We also utilize a statistical technique based on a control chart method, Exponentially Weighted Moving Average (EWMA) for stock behavior anomaly detection [30]. EWMA returns the weighted average of all previous sample means where more recent samples are weighted more highly on the variance while other control charts use a same weight. The exponentially weighted moving average is defined as:

$$z_i = \lambda x_i + (1 - \lambda) z_{i-1} \tag{2}$$

where x_i is the original time series, $0 < \lambda \leq 1$ is a constant, and $i > 2$. λ is typically set to $0.2 \leq \lambda \leq 0.3$ [30]. We then utilize the EWMA control chart to detect abnormal stock behaviors. The control chart is constructed by plotting z_i versus the sample number i. We calculate the control limits using the following equation:

$$CL = \bar{y} \pm L \frac{S}{\sqrt{n}} \sqrt{\frac{\lambda}{(2-\lambda)} \left[1 - (1-\lambda)^{2i}\right]} \tag{3}$$

where $\bar{y}$ is the sample mean, L is the width of the control limits, S is the standard deviation, and n is the number of samples. We set L to 3, which represents the 99% confidence interval band. If the ticker price at a specific time point falls outside the control limits, it is considered to be an abnormal event.

5.3 Sector Average Comparison

We use an anomaly detection algorithm considering group movement [23]. The assumption of this approach is that stocks in the same sector (e.g., IT, oil or airline industries) have a similar movement pattern to each other and the pattern is re-occurring unless there is a critical event affecting on an individual stock (e.g., insider-trading). We compute anomaly scores of stocks by correlating a stock price to the average stock price of the sector that includes the target stock. For example, if the average stock price increases but the stock prices decreases at a specific day, we consider this situation as an abnormal situation.

6 Case Studies

To validate and demonstrate our system, we present several use cases in the technology and airline sectors.

6.1 Technology Sector: Hewlett-Packard Anomaly-Driven Analysis

Here, we describe a hypothetical scenario in which a financial analyst is using our visual analytics system to investigate anomalies and behaviors in the technology sector of the market. The analyst is particularly interested in evaluating the stock price behavior of the company Hewlett-Packard (HP). The analyst begins by examining price data and anomalies for 12 different tickers in the technology sector from August to October, 2011 using the time series view. This is shown in Figure 3. Different parts of this figure have been annotated to facilitate the discussion for the reader. The analyst examines anomalies in the ticker prices for HP using the anomaly detection method described in Section 5. This method accounts for the seasonal variations of each signal for detecting anomalies.

The analyst notices a negative anomaly that occurred on August 19 (seen as an unfilled triangle pointing downwards in the time series). In order to investigate this anomaly further, the analyst applies company-based filtering in the topic stream view (Section 4.5). The results of this operation are shown in Figure 4 (B). He notices a spike in news stories for HP around August 19 in the Topic Stream view (Figure 4 (B)). The analyst decides to select this peak in the Topic Stream view in order to further explore the news stories related to the company. He finds that HP had decided to shut down their tablet business (Figure 4 (1)) and were also expected to announce a spinoff of their PC business into a new company. This provides a context to the analyst on the stock pricing behavior, and also helps him understand the potential cause for the anomalous price drop. From domain knowledge, the analyst also knows that the company had later decided to abandon the plan of branching their PC division into a new company in late August. Correspondingly, he notices another positive anomaly on October 28, and notes that this detected positive anomaly corresponded with this decision. He also verifies this from the topic stream view for October 28 (Figure 4 (2)).

Upon further examination of the HP stock price signal, the analyst also finds a negative anomaly that occurred on August 8. He looks into the topic stream view for the company on this date, but finds no relevant news stories that can provide contextual information about this anomaly. He notes that the other tickers have similar negative anomalies for this date, which suggests that these may be caused due to an event that had sector wide implications. Upon further investigation,

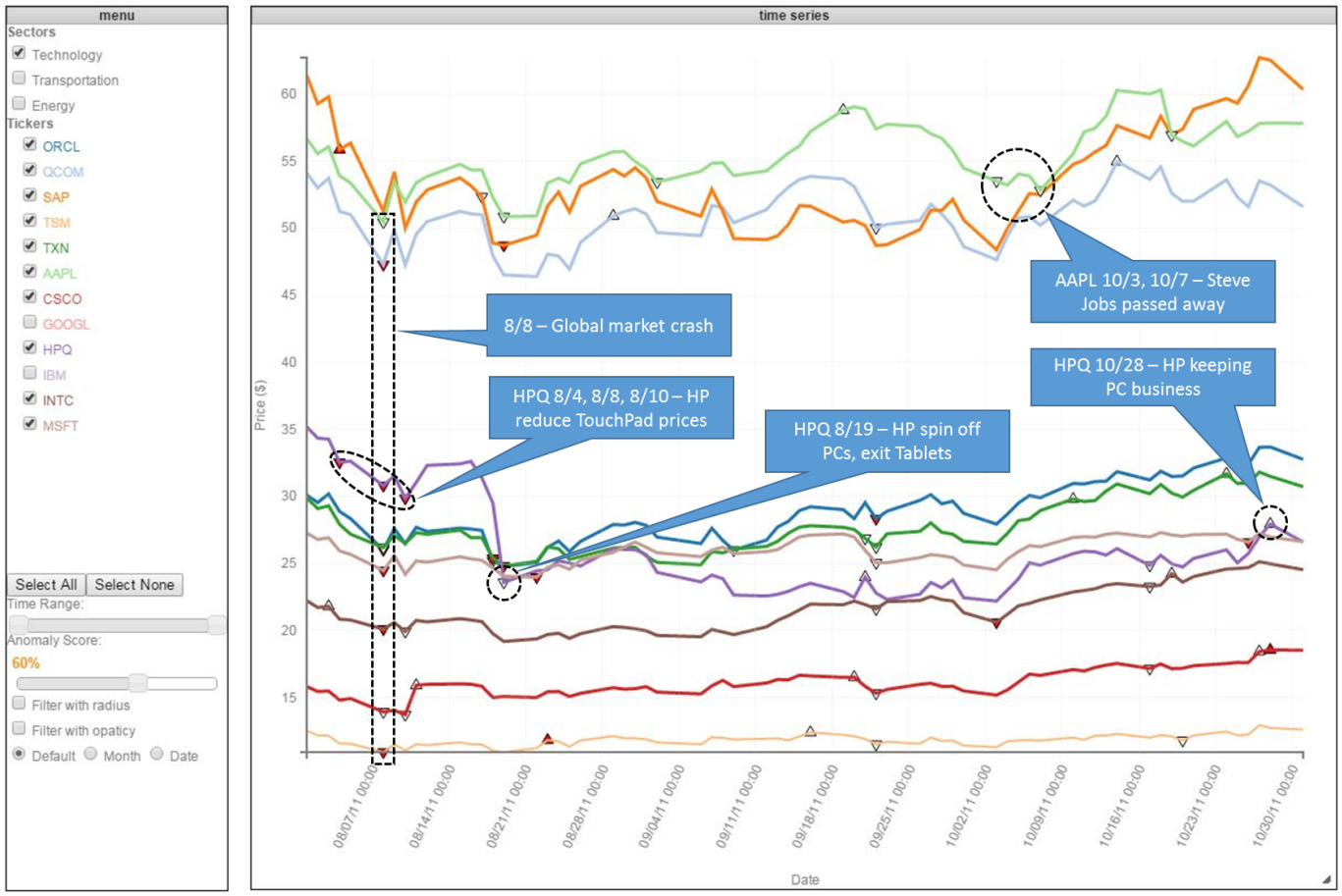


Fig. 3. Time series view showing major technology stock returns and detected abnormal behaviors from August to October, 2011.

the analyst finds that August 8 was the Monday that followed the previous Friday night's credit rating downgrade by Standard and Poor's (S&P) of the U.S. sovereign debt from AAA to AA+[9].This caused a global stock market crash that had ripple effects in the global economy. The analyst notes that this may also have contributed to the anomaly detected on August 8.

Additionally, the analyst notes two other negative anomalies (on August 4 and August 10) around the anomaly that occurred on August 8. After examining the news stories pertaining to the company for these three days, he notes an underlying theme where HP was offering substantial price reductions on their TouchPad tablets after announcing that they would discontinue all their webOS devices. This further provides context behind the anomalies and ticker behaviors.

6.2 Technology Sector: Anomalies in Pricing of Apple Ticker

We now provide an example in which an analyst is performing anomaly driven analysis for Apple stocks. Here, the analyst is interested in exploring the effects of the death of Steve Jobs (October 5, 2011), the founder and then CEO of Apple on the stock price of the company. She observes two anomalies from the time series graph view on October 3rd and 7th, 2011 (highlighted in Figure 3). In order to investigate this further, she uses the topic stream view using the LDA topic modeling technique (Section 4.5) and finds that Topic 18 contains 'steve jobs'(Figure 4 (C)). As expected, she notices news stories related to the death of Steve Jobs around this time period. These stories provides her contextual evidence about that anomalous behavior in the time series. Next, the analyst switches over to the company-based filtering mode in the topic stream view in order to filter the news stories by company. The results of this action are shown in Figure 4 (3). She finds that Apple had received pre-order sales for the newer model of the iPhone 4S around the same period. This is also reflected from the time series view (Figure 3), where she notes that the Apple stock price increased sharply after October 7th.

She also looks into the horizon graph view in Figure 1 (D) and finds that there is a sectorwide trend of decreasing prices in first half of August to October 2011 time period and an increasing trend in the second half. Figure 1 (C) shows the zoomed graph view that makes it easier to explore regions where there is clutter and overlap. The calendar view in Figure 1 (E) shows the distribution of anomalies in the technology sector during this time range.

6.3 Airline Sector Anomaly Analysis

In this example, the analysts perform anomaly driven analysis of ticker data relevant to the airline industry for February and March, 2015. They begin by analyzing the time series anomaly view (Figure 5) by selecting seven airline tickers. They first explore the Lufthansa airline (LHA) ticker price data and find a negative anomaly on March 24th (highlighted in the figure). They analyze this anomaly further by using company-based filtering to explore the relevant news articles. As can be seen from Figure 6 (B), they find that there are a large number of articles about a plane crash for this airline. These articles correspond to the Germanwings, Lufthansa airplane crash[10]that killed all 144 passengers and crew. Further, the analysts explore news articles from 1-2 days before this crash and find that the pilots of Lufthansa

[9]Standard and Poor's, http://www.adn.com/article/obamatriescalm-investors,(accessed March 25, 2015)

[10]http://www.nytimes.com/2015/03/25/world/europe/germanwingscrash.html,(accessed March 25, 2015)

Fig. 4. Topic Stream View (company based filter) (A) and Topic Stream View (LDA Topic Modeling) (B). HP Anomaly related news stories August 10, 2011 (1). HP Anomaly related news stories October 28, 2011 (2). Apple anomaly related news articles October 7, 2011 (3). News stories related to the detected anomalies are highlighted with the red boxes.

Airline were on a 12 hour strike before this crash. The analysts conclude that these two events could be the cause behind the price drop in Lufthansa Airline stocks.

Next, the analysts analyze the American Airlines stock and explore a positive anomaly on March 17th, 2015. They find that American Airlines shows a significant rise in its stock price compared to other tickers in this sector. This can also be observed from the right most quarter of the horizon graph view (Figure 6 (C)). They also observe that most of the airlines in this sector are performing well over this time period. The topic stream view for American Airlines for March 17 shows articles that discuss the airline being added to the prestigious S&P 500 club, and that it had recovered well after its bankruptcy (Figure 6 (2)). These articles help explain the detected positive anomaly.

The analysts continue examining the airline sector data to look for unusual fluctuations in the airline ticker data. They explore three other anomalies detected in the stock prices for UAL, DAL and JBLU that occurred on Feb 19th and 20th (highlighted as rectangle in Figure 5). After examining the related contextual information, they find that these airlines had introduced new baggage fees around these dates. They also investigate a positive anomaly in the Singapore Airlines time series on March 15th, and find that Singapore Airlines was in talks with Jeju air for a potential stake buy. Finally, they also explore a negative anomaly in stock price data for SkyWest Airlines (Figure 5). They find only one news story that discusses a fourth quarter loss suffered by this airline. However, no explanations for this quarterly loss were found from the news sources. Consequently, these cases require analysts to further investigate other data sources to discern the cause of the anomaly.

7 Domain Expert Feedback

Our system was assessed by our collaborating financial analysts who provided us with initial feedback on the system. Overall, the analysts provided positive feedback regarding the ability of the system to explore financial market data and analyze anomalies. In particularly,

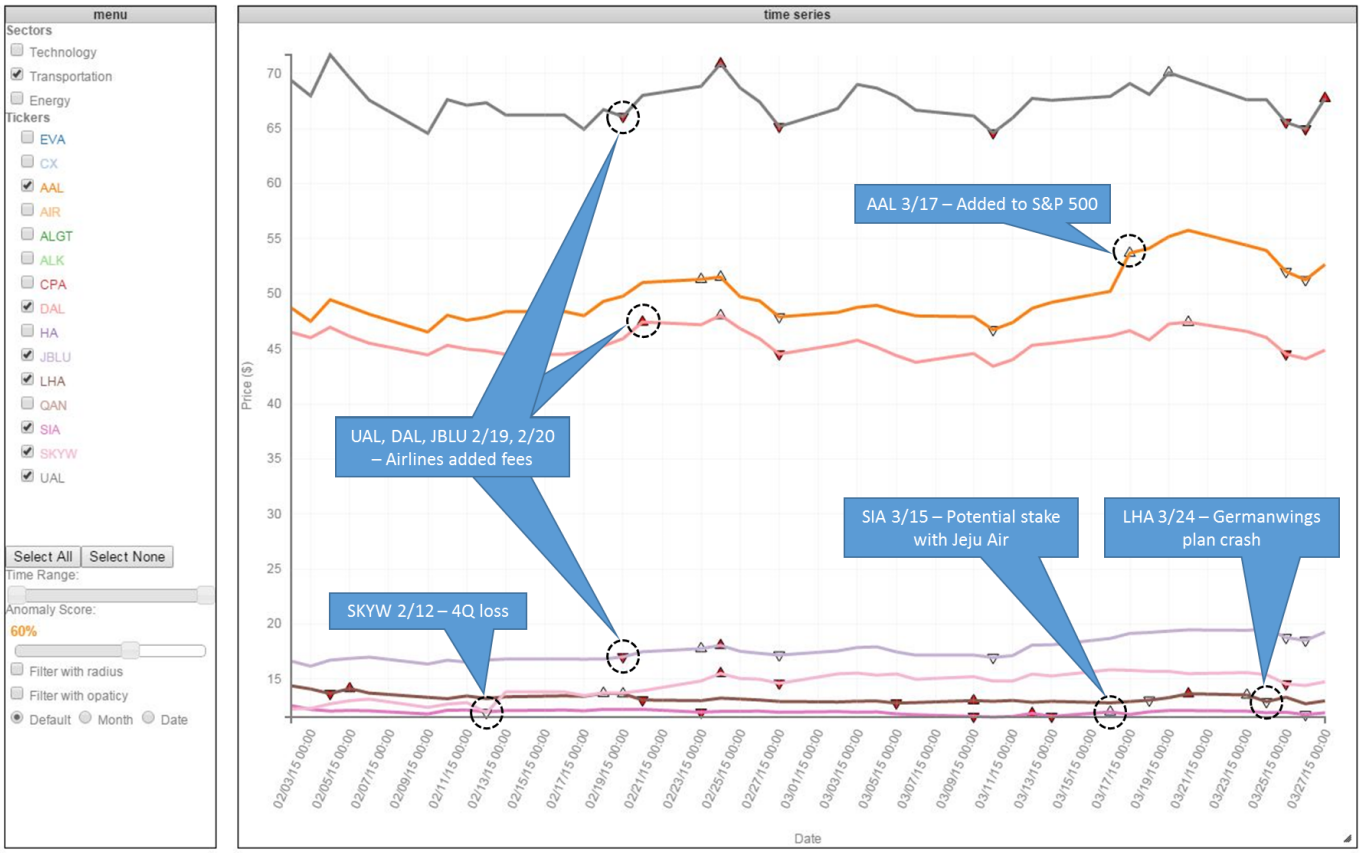


Fig. 5. Time series view showing airline stock prices and detected abnormal behaviors from February to March, 2015.

the analysts found it advantageous that our system utilized automated anomaly detection methods and provided them with auxiliary, but critical, contextual information in order to investigate the anomalies (e.g., news stories). Additionally, the analysts valued the ability to obtain contextual information from news sources on demand for a selected time period, irrespective of whether an anomaly was detected or not. This enabled them to explore news stories and changes taking place around time periods of interest. The analysts particularly liked our linked-view approach where the selection of an entity (e.g., a particular ticker) with the filtering options provides (e.g., country, industry) automatically selection of the relevant features in the other views. We also received positive feedback on the horizon graph. The analysts noted that the horizon graph allowed them to quickly discern sector trends and correlate them with detected anomalies. The Stack Zoom view helped them compare time-series signals across different temporal regions.

We received several additional suggestions from analysts. One suggestion was to provide the ability to easily generate and export reports that contained findings derived from the system. Another recommendation was to provide the ability to import other financial datasets and indicators of the economy in the system (e.g., gross domestic product (GDP), unemployment rates, credit swaps) in order to provide further contextual information around events of interest. Visualizing contextual information in geospace was also suggested by a few analysts. However, they were uncertain about the data that could be shown on the map given the complex global interrelationships of the financial markets. Some suggested providing the ability to filter by geospace in order to test for regional hypotheses; while others suggested using the map view to show the locations of companies and other contextual information on the map (e.g., word clouds on maps). We leave incorporating these suggestions as future work.

8 Conclusions and Future Work

We have introduced a new visual analytics tool for exploring financial data incorporating multiple tightly integrated interactive visualizations. Our time-series view displays stock prices with auto-detected anomalies, while linked views improve awareness acquisition of global and relevant information for anomaly-driven data exploration. We presented three use-case examples using financial data to illustrate the use and potential of the system. Our system can be easily applied to analysis with any other types of financial data. In the future, we plan to incorporate a map view with advanced selection and filtering options to help analysts develop and answer hypotheses. We also plan to deploy our system with our partners and start a longitudinal study.

9 Acknowledgements

The authors would like to thank Christina Ray (Omnis, Inc.), Rick Lawrence (IBM Research) and Robert Kerr (Haystax Technology) for their valuable suggestions and feedback in completing this work.

References

[1] C. Abras, D. Maloney-krichmar, and J. Preece. User-centered design. In *Bainbridge, W. Encyclopedia of Human-Computer Interaction*. Sage Publications, 2004.

[2] J. Alsakran, Y. Zhao, and X. Zhao. Visual analysis of mutual fund performance. In *Proceedings of 13th International Conference on Information Visualisation*, pages 252–259, 2009.

[3] J. Alsakran, Y. Zhao, and X. Zhao. Tile-based parallel coordinates and its application in financial visualization. *Proceedings of Visualization and Data Analysis*, 7530:753003–753003–12, 2010.

[4] H. Y. Aly, S. M. Mehdian, and M. J. Perry. An analysis of day-of-the-week effects in the egyptian stock market. *International journal of business*, 9(3):302–308, 2004.

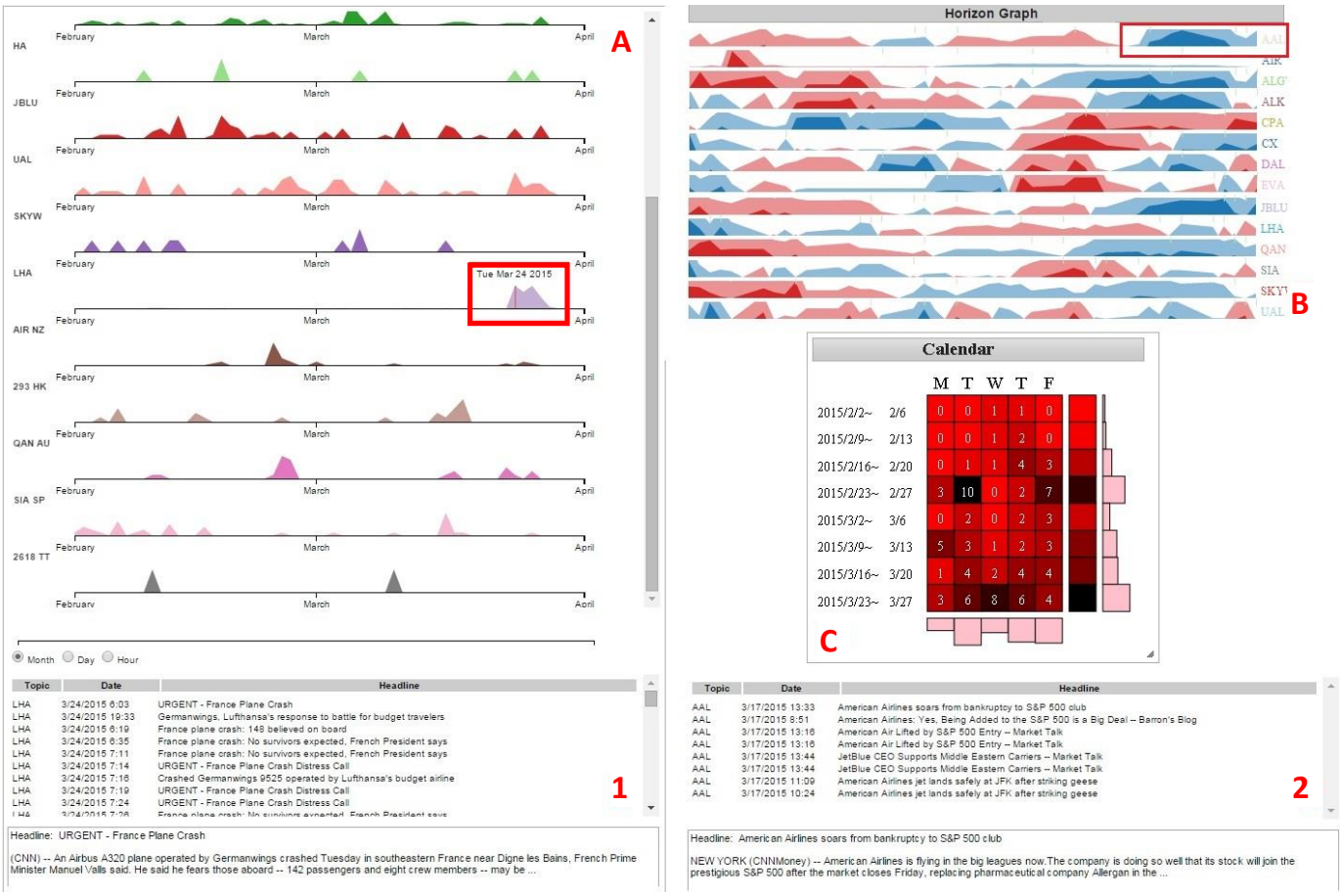


Fig. 6. Topic Stream View (company based filter) (A), Horizon Graph (B), and Calendar View (C). Lufthansa airlines anomaly related stories (1) and American airlines anomaly related stories (2). The highlighted red box indicates an anomaly case of Lufthansa airline.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[6] D. Brodbeck, M. Chalmers, A. Lunzer, and P. Cotture. Domesticating bead: adapting an information visualization system to a financial institution. In *Proceedings of IEEE Symposium on Information Visualization*, pages 73–80, 1997.

[7] M. Cao and J. Wei. Stock market returns: A note on temperature anomaly. *Journal of Banking & Finance*, 29(6):1559–1573, 2005.

[8] R. Chang, M. Ghoniem, R. Kosara, W. Ribarsky, J. Yang, E. Suma, C. Ziemkiewicz, D. Kern, and A. Sudjianto. Wirevis: Visualization of categorical, time-varying data from financial transactions. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology*, pages 155–162, Oct 2007.

[9] R. Chong, R. Hudson, K. Keasey, and K. Littler. Pre-holiday effects: International evidence on the decline and reversal of a stock market anomaly. *Journal of International Money and Finance*, 24(8):1226–1236, 2005.

[10] T. chung Fu. A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1):164–181, 2011.

[11] R. B. Cleveland, W. S. Cleveland, J. E. McRae, and I. Terpenning. Stl: A seasonal-trend decomposition procedure based on loess (with discussion). *Journal of Official Statistics*, 6:3–73, 1990.

[12] T. Dang and V. Lemieux. A functional framework for evaluating financial visualization products. In *Financial Analysis and Risk Management*, pages 115–153. Springer Berlin Heidelberg, 2013.

[13] R. Edwards, J. Magee, and W. Bassetti. *Technical Analysis of Stock Trends*. AMACOM, 2007.

[14] N. Elmqvist, T.-N. Do, H. Goodell, N. Henry, and J. Fekete. Zame: Interactive large-scale graph visualization. In *Proceedings of IEEE Pacific Visualization Symposium*, pages 215–222, 2008.

[15] N. Elmqvist, P. Dragicevic, and J.-D. Fekete. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1539–1148, 2008.

[16] M. D. Flood, V. L. Lemieux, M. Varga, , and B. L. W. Wong. The application of visual analytics to financial stability monitoring. *Social Science Research Network, SSRN Scholarly Paper ID 2438194*, May 2014.

[17] K. R. French. Stock returns and the weekend effect. *Journal of financial economics*, 8(1):55–69, 1980.

[18] M. R. Gibbons and P. Hess. Day of the week effects and asset returns. *Journal of business*, pages 579–596, 1981.

[19] P. J. Groenen and P. H. Franses. Visualizing time-varying correlations across stock markets. *Journal of Empirical Finance*, 7(2):155–172, 2000.

[20] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han. *Outlier Detection for Temporal Data*. Synthesis Lectures on Data Mining and Knowledge Discovery. Morgan & Claypool Publishers, 2014.

[21] M. A. Harrower and C. A. Brewer. Colorbrewer.org: An online tool for selecting color schemes for maps. *Cartographic Journal*, 40(1):27–37, 2003.

[22] C. C. Holt. Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting*, 20(1):5–10, 2004.

[23] H. Janetzko, F. Stoffel, S. Mittelstädt, and D. A. Keim. Anomaly detection for visual analytics of power consumption data. *Computers & Graphics*, 38:27–37, 2014.

[24] W. Javed and N. Elmqvist. Stack zooming for multi-focus interaction in time-series data visualization. In *Proceedings of IEEE Pacific Visualization Symposium*, pages 33–40, 2010.

[25] D. A. Keim, M. C. Hao, J. Ladisch, M. Hsu, and U. Dayal. Pixel bar charts: A new technique for visualizing large multi-attribute data sets without aggregation. In *Proceedings of IEEE Symposium on Information Visualization*, pages 113–120, 2001.

[26] D. A. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann. *Mastering The Information Age-Solving Problems with Visual Analytics*. Florian Mansmann, 2010.

[27] S. Ko, R. Maciejewski, Y. Jang, and D. S. Ebert. Marketanalyzer: An interactive visual analytics system for analyzing competitive advantage using point of sale data. *Computer Graphics Forum*, 31(3):1245–1254, 2012.

[28] V. L. Lemieux, B. Fisher, and T. Dang. *The Visual Analysis of Financial Data*. Cambridge University Press, 2014.

[29] C. S. Merino, M. Sips, D. A. Keim, C. Panse, and R. Spence. Task-at-hand interface for change detection in stock market data. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, pages 420–427. ACM, 2006.

[30] D. C. Montgomery. *Introduction to statistical quality control*. John Wiley & Sons, 2007.

[31] J. Murphy. *Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications*. New York Institute of Finance, 1999.

[32] S. Rudolph, A. Savikhin, and D. Ebert. Finvis: Applied visual analytics for personal financial planning. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology*, pages 195–202, 2009.

[33] T. Saito, H. N. Miyamura, M. Yamamoto, H. Saito, Y. Hoshiya, and T. Kaseda. Two-tone pseudo coloring: Compact visualization for one-dimensional data. In *Proceedings of IEEE Symposium on Information Visualization*, pages 173–180, 2005.

[34] A. Savikhin, H. C. Lam, B. Fisher, and D. Ebert. An experimental study of financial portfolio selection with visual analytics for decision support. In *Proceedings of 44th Hawaii International Conference on System Sciences*, pages 1–10, 2011.

[35] A. Sawant. Stockviz: Analyzing the trend of stocks in major auto, oil, consumer, and technology companies. In *Proceedings of the 2009 International Conference on Modeling, Simulation & Visualization Methods*, pages 278–284, 2009.

[36] T. Schreck, J. Bernard, T. Von Landesberger, and J. Kohlhammer. Visual cluster analysis of trajectory data with interactive kohonen maps. *Information Visualization*, 8(1):14–29, Jan. 2009.

[37] T. Schreck, T. Tekušová, J. Kohlhammer, and D. Fellner. Trajectory-based visual analysis of large financial time series data. *SIGKDD Explorations Newsletter*, 9(2):30–37, Dec. 2007.

[38] G. W. Schwert. Anomalies and market efficiency. *Handbook of the Economics of Finance*, 1:939–974, 2003.

[39] E. Sorenson and R. Brath. Financial visualization case study: Correlating financial timeseries and discrete events to support investment decisions. In *Information Visualisation (IV), 2013 17th International Conference*, pages 232–238, July 2013.

[40] D. P. Tegarden. Business information visualization. *Communications of the AIS*, 1:2–38, 1999.

[41] T. Tekušová and J. Kohlhammer. Applying animation to the visual analysis of financial time-dependent data. In *Proceedings of International Conference on Information Visualization*, pages 101–108, 2007.

[42] J. J. Thomas and K. A. Cook, editors. *Illuminating the Path: The R&D Agenda for Visual Analytics*. IEEE Press, 2005.

[43] F. Wanner, A. Stoffel, D. Jckle, B. C. Kwon, A. Weiler, and D. A. Keim. State-of-the-art report of visual analysis for event detection in text data streams. In *EuroVis - STARs*, pages 125–139. The Eurographics Association, 2014.

[44] M. Wattenberg. Visualizing the stock market. In *Extended Abstracts on Human Factors in Computing Systems*, pages 188–189. ACM, 1999.

[45] J. V. Wijk and E. V. Selow. Cluster and calendar based visualization of time series data. In *Proceedings of IEEE Symposium on Information Visualization*, pages 4–9. IEEE, 1999.

[46] E. Wu and P. Phillips. Financial markets in motion: Visualising stock price and news interactions during the 2008 global financial crisis. *Procedia Computer Science*, 1(1):1765–1773, 2010.

[47] H. Ziegler, M. Jenny, T. Gruse, and D. Keim. Visual market sector analysis for financial time series data. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology*, pages 83–90, 2010.

[48] H. Ziegler, T. Nietzschmann, and D. Keim. Visual exploration and discovery of atypical behavior in financial time series data using two-dimensional colormaps. In *Proceedings of 11th International Conference on Information Visualization*, pages 308–315, July 2007.

[49] H. Ziegler, T. Nietzschmann, and D. A. Keim. Visual analytics on the financial market: Pixel-based analysis and comparison of long-term investments. In *Proceedings of 12th International Conference on Information Visualization*, pages 287–295, 2008.

DemographicVis: Analyzing Demographic Information based on User Generated Content

Wenwen Dou, Isaac Cho, Omar ElTayeby, Jaegul Choo, Derek Xiaoyu Wang and William Ribarsky

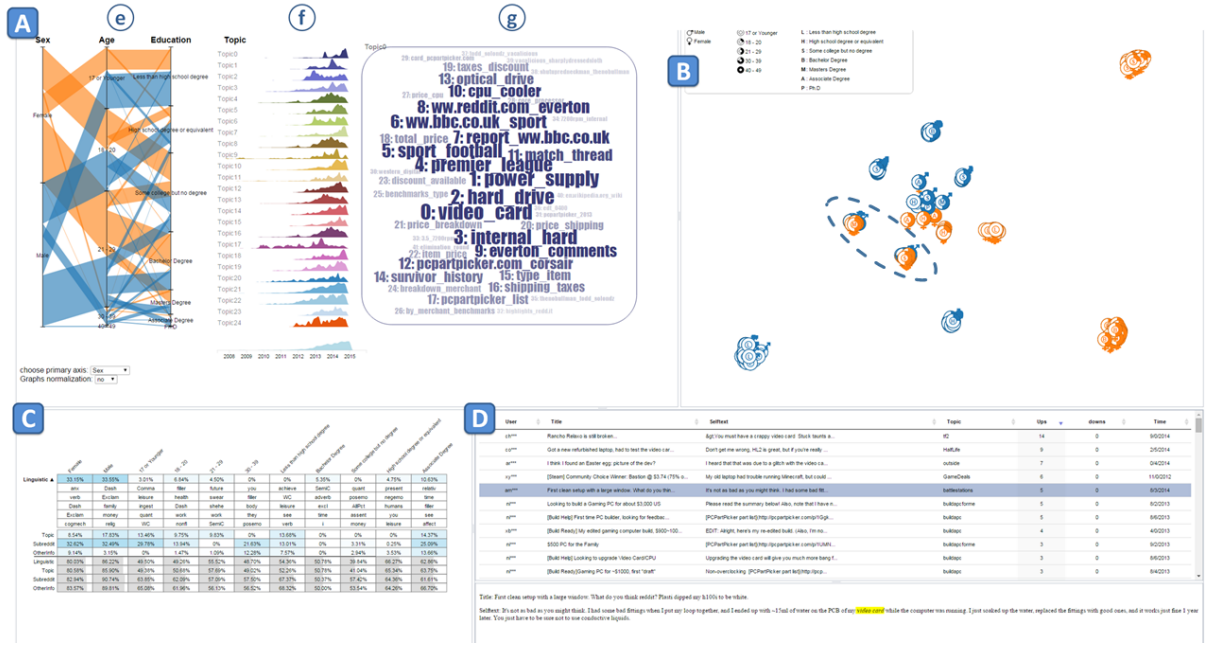


Figure 1: DemographicVis interface: A) Parallel Sets with word cloud view that connects demographic groups to user generated content, B) user cluster view that groups users based on topic interests, C) feature ranking view that presents the predicative power of various features, D) posts view showing details on demand.

Abstract

The wide-spread of social media provides unprecedented sources of written language that can be used to model and infer online demographics. In this paper, we introduce a novel visual text analytics system, DemographicVis, to aid interactive analysis of such demographic information based on user-generated content. Our approach connects categorical data (demographic information) with textual data, allowing users to understand the characteristics of different demographic groups in a transparent and exploratory manner. The modeling and visualization are based on ground truth demographic information collected via a survey conducted on Reddit.com. Detailed user information is taken into our modeling process that connects the demographic groups with features that best describe the distinguishing characteristics of each group. Features including topical and linguistic are generated from the user-generated contents. Such features are then analyzed and ranked based on their ability to predict the users' demographic information. To enable interactive demographic analysis, we introduce a web-based visual interface that presents the relationship of the demographic groups, their topic interests, as well as the predictive power of various features. We present multiple case studies to showcase the utility of our visual analytics approach in exploring and understanding the interests of different demographic groups. We also report results from a comparative evaluation, showing that the DemographicVis is quantitatively superior or competitive and subjectively preferred when compared to a commercial text analysis tool.

Keywords: Visual Text Analysis, user interface, social media, demographic analysis

Index Terms: H.5.2 [Information interface and presentation (e.g., HCI)]: User Interfaces—Graphical user interfaces (GUI)

1 Introduction

Demographic analysis provides valuable insights on social, economic, and behavioral issues. On a macro level, analyzing demographic information sheds light on a range of future economic factors from gross domestic product growth and inflation to interest rates [10]. On a micro level, demographic analysis yields valuable information on businesses, communities, and other aspects that are closely related to our daily lives. From a business-oriented point of view, understanding demographics is important for business development, marketing, and customer relationship management. Businesses market products or services through targeted approaches to different segments of the population, which are often identified by demographic analysis. Regarding issues that are more specific to the internet era, analyzing demographic information could help study issues such as online privacy and security. A recent study on phishing susceptibility among different demographics groups has identified several factors that influence users' online behaviors [24].

While demographic information is traditionally collected through census and surveys, the abundance of user-generated content from social media and weblogs provide a unique opportunity for inferring demographic information directly based on users' input. Traditional demographic analysis is usually time-consuming and costly, especially if the survey needs to cover a large population. But with the help of social media, a fast and direct channel can be established with individuals for demographic surveys. In fact, researchers have taken advantage of the channel to collect demographic data through social media including Facebook [23, 26, 4] and Twitter [20, 5]. Various research methods have been developed to analyze the collected data, in order to identify features that can be used to predict demographic information such as age and gender.

Individual users post online discussions regarding their daily lives, international and local events, and other topics of interests. There is an opportunity to establish a direct connection between demographic groups and topics of interests, language style, as well as online social behavior. Such connection provides important insights on users interests, social and behavioral patterns, and internet cultures, possibly distinguished by different demographic groups.

Much of the previous research on demographics analysis can be organized into two categories. Research in the first category analyzed user-generated content through counting word usage over pre-determined categories of language in order to distinguish demographic groups [3, 7, 17]. Such approaches yield language usage features that are easy to interpret and can be used to make sense of the commonalities and differences among different groups. The second category of research adopted a more "open vocabulary" approach [23, 5, 16], which does not restrict the analysis to *a priori* word categories. Instead, all words from user generated content can be used as features to classify users into different demographic groups. Such methods employ machine learning algorithms including SVM and Naive Bayes for age and gender classification. The objective of these approaches in the second category is to experiment with different features and optimize the machine learning algorithm to achieve the best accuracy for predicting demographic factors. In contrast to the first category, producing meaningful and interpretable results that distinguish demographic groups is not the focus of these methods. As a result, one drawback is that the features that distinguish the demographics group are not in a form that can be consumed by interested parties.

In this paper, we offer DemographicVis[1], a visual analytics approach to demographic analysis that combines the merits of the above two categories of research, in that we take a data-driven perspective and we establish a direct link between demographic groups and meaningful, easy-to-interpret features. More importantly, we provide an interactive visual interface for users to make sense of the connection between demographic groups and features that distinguish them, including topical and linguistic features. Compared to previous studies and computational methods on demographic analysis, the novelty of DemographicVis is that it enables interested parties to directly connect demographic information with the computationally extracted yet meaningful features of the corresponding demographic groups. Previous text visualization systems, such as [9, 27, 8, 2], focus on developing novel approaches to explore and analyze large corpora alone. In contrast, DemographicVis explicitly connects categorical data (demographic factors in this case) with textual contents.

The major contributions of the papers include:

- A new visual analytics system, DemographicVis, is presented that integrates state-of-the-art analytical methods with a novel visual interface to clearly show the relationship between demographic information and user-generated content. The visual interface includes a rotated Parallel Set and interactive word clouds, and is tailored to present the connection between demographic information and the features that distinguish various demographic groups. DemographicVis makes explicit connection between categorical data (demographic information) and textual data (user generated content).
- DemographicVis enables a transparent way to conduct demographic analysis, making the features that best describe different demographic groups easy to interpret and ready to consume by the end users. Topical, linguistic, and peripheral features are extracted from both user-generated contents and meta data using multiple machine learning algorithms. The features are also ranked to demonstrate their importance for predicting demographic information.
- A quantitative evaluation is provided to compare DemographicVis to SAS TextMiner, a commercial text mining software for extracting insights from textual data. The evaluation results show that DemographicVis received significantly higher rating in terms of ease of use and ease of learning with comparable performance on achieving various tasks.

[1] http://demographicvis.uncc.edu

2 Related Work

Demographic analysis has been an important research domain. Researchers from Social Sciences gained psychological insights through studies that link language use with age and gender, while researchers from Computer Science have focused on introducing and improving algorithms to predict demographic information.

2.1 Linguistic analysis on age and gender

The typical approach of correlating age and gender with language use involves counting word usage over a priori word-categories. The most commonly used word-category lexicon is the Linguistic Inquiry and Word Count (LIWC) dictionary. Several studies have leveraged LIWC and focused on function words to study age and gender. Research by Chung et al. [7] and Argomon et al. [3] on gender analysis found that males use more articles, while females use more first-person singular pronouns. Also focusing on examining function words, Newman et al. reported several findings [17]. First, women use more certainty words while men tend to have greater use of numbers, articles, long words, and swearing. Second, women were more likely than men to refer both to positive feelings and to negative emotions, specifically, sadness and anxiety. Third, strong evidence was found that women seem to have more of a "rapport" style, discussing social topics and expressing internal thoughts and feelings more often, whereas men "report" more often, describing the quantity and location of objects.

As of age, through linking language use and aging, Pennebaker et al. [19] found that with increasing age, individuals use more positive and fewer negative affect words, use fewer self-references, more future-tense and fewer past-tense verbs. In the context of blogging, Schler et al. [22] identified a clear pattern of differences in content and style: regardless of gender, writing style grows increasingly "male" with age: pronouns and assent/negation become scarcer, while prepositions and determiners become more frequent.

We see our visual analytics approach as complimentary to the linguistic studies. Through coupling the semantically meaningful topics and relationships between demographic groups identified in our approach, with the general patterns identified by linguistic studies, higher order thought patterns can be revealed and outcomes can be solidified and become more interpretable.

2.2 Classification models for predicting age and gender

Computer Scientists have also used linguistic features to build and improve models that predict age or gender. Examining information from social media users, Burger et al. [5] experimented with Support Vector Machines, Naive Bayes, and Balanced Winnow2

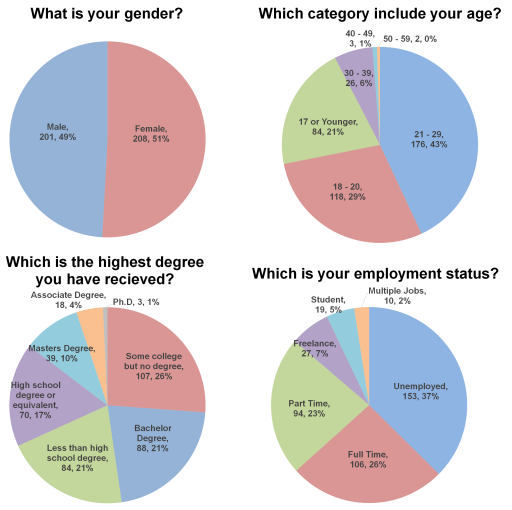


Figure 2: Demographic distribution on gender, age, education, and employment status based on the collected data.

[14] to build classifiers to predict gender. Descriptions for Twitter user such as screen name and full name are used in addition to tweets to improve the accuracy of the classifier. Rao et al. [20] introduced stacked-SVM-based classification algorithms over a rich set of features to classify gender, age, regional origin and political orientation, while Schler et al. [22] leveraged both style-based and content-based features to classify age and gender for thousands of bloggers. To improve the accuracy on gender classification, Mukherjee et al. [16] proposed techniques for new feature selection in addition to commonly used word classes, unigram, and ngram. Empirical evaluation of the approach improves the classification accuracy of gender for blog authors.

Comparing to the linguistic analysis research, the above mentioned classification approaches focus more on predicting age and gender, and less on gaining psychological insights from analyzing the language use of different demographic groups. As a result, interpretable results that distinguish demographics groups are difficult to obtain from the classification models.

3 Data Collection

3.1 Demographic information collection

To collect demographic information from online users, we designed and posted a survey on Reddit.com, an online link-sharing community and message board. Reddit has gained popularity in recent years. In order to obtain demographic information directly from this community, we first compiled a set of multiple choice questions. The subreddit that allowed us to post the survey is r/SampleSize. This community is dedicated to generating and answering surveys. In our survey, we also designed a set of control questions with simple answers to rule out the participants who answer the survey questions randomly.

The information collected through the survey includes each responder's gender, gender expression, age group, education, current location, income level, religious affiliation, etc. 482 users participated in our survey, 409 users were included in the final data collection after filtering based on the control questions. Figure 2 presents the summary of information all 409 responses. The summary suggests that our pool of participants are fairly balanced as to gender, although Reddit is thought to be a male-dominant community. In terms of age group, the results showed a good coverage of individuals ranging from 17 or younger to 39 years old.

Note that previous studies that focus on analyzing and predicting just age and gender tend to have larger sample population, since gender and age information is more readily available. However,

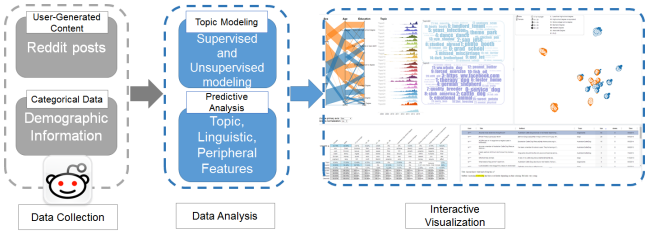


Figure 3: System architecture of DemographicVis. Section 3, 4, and 5 introduce the Data Collection, Data Analysis, and Interactive Visualization component respectively.

in our study, we collected more detailed demographic information well beyond age and gender. Although a sample of 409 redditors may not provide sufficient statistical power for generalizing our findings to broader contexts, our visual analytics approach to demographic analysis can be applied to information collected from a much larger population.

3.2 Collection of user-generated content

The objective of our research is to connect demographic information with the content the users posted on social media through visual analytics means. To this aim, we collected the posts from the 409 valid users of our survey. We developed a python crawler to gather the posts of this group of users through Reddit's public API. 169,707 posts were included in the final dataset, the time stamp of posted comments ranged from 2011 to date. Unlike Twitter, the posts on Reddit do not have length restriction. Therefore a large portion of the comments contained at least a few sentences.

In our final dataset, each record is one post from an individual user. Since each user belongs to a certain demographics group, each record is then associated with the corresponding demographic group. Different from previous studies, we analyze the user-generated content based on multi-attribute demographic groups as opposed to examining attributes such as age, gender, ethnicity individually. Therefore, each user comment in our database is tagged with one multi-attribute demographic group. The attribute combination can be chosen based on the analysis needs. For instance, one common combination is gender, age and education. An example of a particular demographic combination could be {Female, Age 18 - 20, High school degree}.

4 DemographicVis: A Visual Analytics Approach for Demographic Analysis

Our approach combines analytical methods and an interactive visual interface to enable the analysis of the relationship between demographic information and user-generated content. The system architecture of DemographicVis is shown in Figure 3. In this section, we focus on introducing the "Data Analysis" component; the "Interactive Visualization" will be described in the next section.

4.1 Data modeling and feature analysis

To describe different demographic groups based on user generated content, we extract features from the reddit posts, including linguistic and topic features. We also extract additional features from the meta-data associated with each post, and use them in conjunction with topic and linguistic features for predictive analysis.

4.1.1 Topic features for describing demographic groups

To describe the demographic groups based on user generated content, a concise and meaningful summary of interests of each individual demographic group needs to be extracted. To this aim, we perform supervised topic modeling to extract topics for each demographic group. Since the direct relationship between topics and

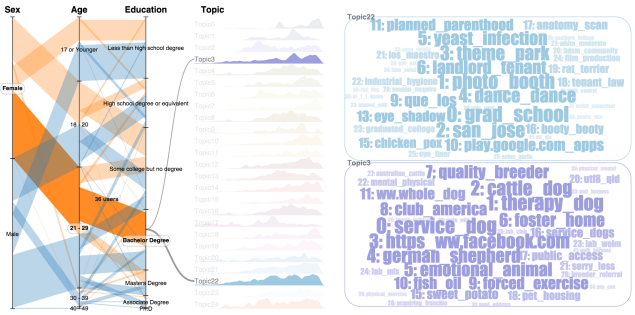


Figure 4: Hovering mouse over demographic group "Female, 21-29, Bachelor degree" leads to the highlighting of the two topics that summarize the interests of the group.

the demographic group is essential to our objective, we want to establish direct links during the modeling process. To incorporate demographic information directly into the topic extraction process, we adopt the Tag-LDA model [15] that was designed to include tags or labels of each document during the topic modeling process. In our approach, each multi-attribute demographic group serves as a tag for each comment a user posted on Reddit. 36 unique multi-attribute demographic groups are found in our data. As a result, we can now use the topic results to describe the interests of each demographic group. For instance, as seen in figure 4, the two topics that best describe multi-attribute group "Female 21-29 with Bachelor degree" are Topic 22, which includes keywords "grad_school, study_abroad, theme_park", and Topic3 that focuses on discussing dogs and puppies.

When performing the topic extraction, we employ a bigram approach that treats two consecutive words in a document as a unit of analysis. According to [12], bigrams serve as better feature representations compared to unigrams. Employing bigrams during topic modeling enables us to discover more phrases with richer meaning such as "birth control" and "medical marijuana".

4.1.2 Topic, linguistic, and peripheral features for predicting demographic factors

Topic features are great for presenting a visual summary of the interest of different demographic groups. At the same time, topic features can also be used for predicting demographic information based on user generated content. For the task of predictive analysis, we extract peripheral features and further analyze which feature or combination of features can be used to best predict demographic information (e.g. age, gender, education, etc.). This is especially useful given that the availability of demographic information on social media is scarce and often unreliable. In this section, we describe features we extracted from both user generated content and meta data for the purpose of predictive analysis. These features are then ranked and presented in the visualization so that the users can interactively make sense of how each feature contributes to the task of predicting different demographic attributes.

Our entire feature set contains three subsets of features: topic proportions, linguistic features, and peripheral features extracted from metadata. It would be convenient to reuse the topic proportions from the above supervised topic modeling process, however, we choose not to because linking the labels with the test data that may not have a good coverage on all content will lead to a poor overall prediction performance. In the following descriptions, we will introduce how we derive the features for predictive analysis.

Topic proportions. To obtain the topic proportions, we first construct a term-document matrix using a bag-of-words model based on all available reddit posts. Next, we perform topic modeling using a method called nonnegative matrix factorization [6] with the total number of topics set at 100. We then obtain a 100-dimensional topic-wise vector representing each reddit post. Next, for each user, we sum up these 100-dimension topic-wise vectors of all the reddit posts a user has written, and this aggregated vector works as our topic proportion feature for a particular reddit user.

Linguistic feature. To extract linguistic feature from user generated content, we performed the LIWC analysis, which involves counting word usage over *a priori* word categories. We extracted 82 linguistic variables from the Reddit posts. Such variables include general descriptors, standard linguistic dimensions, etc. A complete list of the variables can be found at *http://www.liwc.net/descriptiontable1.php*. The linguistic feature is then used to perform predictive analysis in conjunction with the topic features.

Peripheral features. Features in this category include the subreddit and other features. We generated the subreddit feature as the 4,296-dimensional vector whose dimension is the total number of unique subreddit categories, where the value represents the count of the reddit text that a user has written in the corresponding subreddit category. The peripheral feature includes a 5-dimensional feature vector containing the total number of reddit posts for a particular user, the ratio of original posts (not including comments on other redditors' posts) to the total number of posts, the total number of thumb-up, the total number of thumb-down received, and the total number of comments that other reddit users have written in response to the user's reddit posts.

4.1.3 Predictive analysis

We obtain a 4,483-dimensional vector for each Reddit user. Then we build a binary classifier by using two groups at a time: one containing users in a particular demographic group and the other containing those in the rest of the groups. Considering the high dimensionality and the heterogeneity of these feature vectors, it is critical to use the most capable learner that can properly handle our data. To this aim, we use a gradient boosting tree (GBtree) [11][2], a state-of-the-art ensemble model that adopts a decision tree as an individual learner. The classification performance is shown as a feature table in Figure 1C.

The feature table is divided into two parts: the top table presents the contribution of different features in predicting the demographic variables, while the bottom table presents the accuracy of the predictive analysis for different demographic variables, measured by the Area Under the ROC Curve (AUC). The reason for choosing the AUC value as our evaluation measure rather than a simpler measure such as the prediction accuracy is because our dataset is highly unbalanced between a particular demographic group and the rest. The AUC value is not dependent on the imbalanceness of a dataset.

The top table shows the variable importance scores when we only incorporate particular features. That is, we perform the prediction experiments by using only one feature group corresponding to each row (e.g., 'Linguistic', 'Topic', etc.) at a time and measure how much the binary classification performance **increases** in terms of the AUC value from a random guess classifier of 0.5.

The bottom half of the table shows the **cumulative** AUC values when we gradually incorporate more features. For example, the first row shows the AUC values only when using the 'Linguistic' features, and the second row shows the AUC values when using the 'Topic' features together with the 'Linguistic' ones. From this table, one can see gender can be well predicted, as shown as the overall performance of 83.57% and 89.81% in the first two columns. Our study is one of the very first ones that provides promising results for predicting diverse demographic characteristics in terms of

[2]The implementation of GBtree we used is available at https://sites.google.com/site/carlosbecker/resources/gradient-boosting-boosted-trees

age and education levels, among other aspects, rather than just predicting gender. More importantly, the use of visualization helps users to make sense of the contribution of different features.

5 VISUAL INTERFACE

The interactive visualization permits the sense making and comparison of different demographic groups, as well as identifying features that can be used for demographic information prediction.

To enable interactive demographic analysis based on the features introduced in Section 3, we designed a web-based visual interface that connects categorical data to user generated content. The interface consists of multiple views that were designed based on tasks summarized from interviewing users who are interested in performing demographic analysis. Such users include our industry partners who are interested in performing demographic analysis for marketing purposes, and academic professors who are interested in understanding online behaviors of different demographic groups.

In the context of connecting demographic information with posts from social media, the interviewees are most interested in the following 3 analysis tasks:

- T1 What are different demographic groups interested in? Do different demographic groups have distinct interests that are reflected in what they post?
- T2 Which demographic groups share similar interests?
- T3 Can we leverage information derived from posts on social media to successfully classify online users into different demographic groups when there is little ground truth available?

To address the three tasks, we introduce an interactive visual interface that consists of the following three views.

5.1 Parallel Sets + Word Cloud: connecting demographic groups to topic features

To address T1, we transform and combine visualizations tailored for categorical data and topic results. Specifically, we combine transformed Parallel Sets with interactive word clouds. Parallel Sets (ParSets) [13] is an effective visualization application for demographic or other categorical data. Multiple word clouds reveal the topical results. In addition, the temporal trends of the topics are also presented in small multiples to take advantage of the temporal resolution provided by social media data.

5.1.1 Parallel Sets for demographic data

ParSets is designed for visualizing relationships between dimensions in categorical data which we apply here to three demographic dimensions: gender, age and educational level. In contrast to the original ParSets layout, we made a design decision to rotate the ParSets by 90 degrees for two reasons: 1. The dimensions are then drawn from left to right, which conforms to the natural reading direction of most people; 2. Such rotation allows easy connection between the demographic dimensions and the topic word clouds as shown in Figure 1A.

To enable users to explore hypotheses regarding different demographic variables in a flexible manner, the DemographicVis interface permits interactive selection of the starting dimension, since the first dimension in ParSets determines coloring and other choices, as described below. Ribbons connecting adjacent dimensions are sized according to the number of users falling under the combination of the two demographic variables. As seen in figure 1A, the label for each dimension is on the top while the label for each category within a dimension is placed at the center of the category. The color of the ribbons is determined by the first dimension, as in the original ParSets. Since we allow users to interactively set the first dimension, colors will be reassigned automatically based on the choice of the first dimension.

User interaction. When the user hovers over a ribbon, the ribbon is highlighted while the other ribbons are dimmed. At the same time, the corresponding demographic variables are highlighted. To enable examination of a certain demographic variable group, clicking on a ribbon will keep the ribbon highlighted when hovering out. This interaction is important when trying to connect demographic groups to their corresponding topics.

5.1.2 Topic representation: Word Cloud + Streamlines

To allow users to understand the topic interests of various demographic groups, we link interactive word clouds and topic streams to the demographic information. Each word cloud depicts one topic derived through the modeling process described in Section 3.1 while each topic stream portrays the temporal trend. The time span of the topic stream ranges from late 2008 to early 2015, with most posts published between 2013 and 2015. Because of the supervised modeling process, each topic describes interests of a specific demographic group. For instance, as shown in figure 1g, topic 0 describes the interests of group "male, 18-20, high school degree or equivalent", which includes keywords related to computer hardware (video_card, hard_drive, etc.) and sports games (premier_league, sport_football). To highlight the importance ranking of keywords in the word cloud, we use a combination of font size, opacity, and numbering. The size of each bigram is determined by the probability of the bigram in a particular topic. To further distinguish the most important bigrams, we added a number in front of the leading bigrams to indicate their precise ranking in the topic. The bigrams are animated when popping up, so that the most probable ones show up first. Each topic is drawn inside a rectangular bounding box, with the size of the box dynamically determined based on the total number of topics to be displayed.

User interaction. Users can reveal relationships between demographic groups and topics via a two-way interaction. On the one hand, hovering the mouse over a certain demographic group would highlight the corresponding topic feature(s) that describe the interests of the demographic group. On the other hand, hovering over a particular topic stream would highlight the demographic group(s) that are interested in this topic. To maintain the connection to topic meanings, the corresponding word clouds are drawn when hovering over either a demographic group or a topic stream.

To help users better understand the topics, clicking on a bigram in a topic brings up a list of posts that contain the bigram. The list of posts will be displayed in the post view shown in Figure 1D. The post view displays information including anonymized user name, the post, the subreddit information, a time stamp, and votes on the posts. Seeing how a bigram is mentioned in the detailed posts helps users to understand certain keywords that might seem obscure at first. For instance, "nurarihyon_mago" appeared as the most important keyword in topic6, clicking on the term leads to posts that discuss the Japanese manga that turns out to be very popular among the group "Female, 17 or younger, Less than high school degree".

5.2 User Cluster View

To address T2, namely to find out whether the demographic groups have distinct or similar interest, we grouped the 409 redditors that participated in our survey based on the content they posted. Such grouping allows one to easily discover whether users belonging to the same demographic group share similar interests.

To generate clusters based on the interests of the users, we leverage the topic results (Section 4.1.1). The similarity between two users are computed by calculating the KL divergence of their topic distributions. To map the similarity matrix computed for all 409 redditors, we leverage a dimensionality reduction method called t-Distributed Stochastic Neighbor Embedding (t-SNE) [25]. t-SNE is particularly well suited for the visualization of high-dimensional

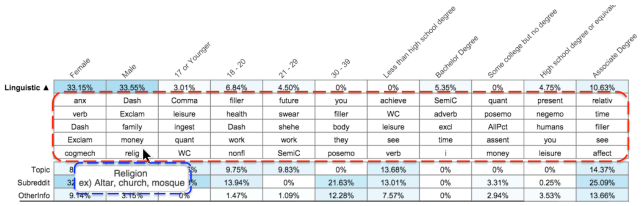


Figure 5: Feature table with linguistic features expanded. Five highest ranked linguistic features are displayed for analysis.

datasets since it generates compact and easily separable clusters [25].

To further assist users in linking the above dimensionality reduction results to demographic information, we designed glyphs to encode the demographic factors in the clusters. With the glyphs, it is easy to see whether users belonging to the same demographic group are clustered together. As seen in Figure 1B, each glyph represents one redditor, with their demographic variables (gender, age, education) captured by the glyph. We went through an iterative process to finalize the glyph design so that it is both intuitive and easy to read. The gender variable is represented by the two standard gender symbols denoting male ♂and female ♀. The age variable (5 age groups) is encoded in the outer ring of each glyph, with each tier in age group adding 1/5 of the filling. Lastly, the education variable is encoded in the inner circle of the glyph as the first letter of the education level. The glyph encodings are shown in the legend. As seen in figure 1B, most bigger clusters, especially the ones located on the outer parts of the view, mainly contain one demographic group. Such observation leads to the hypothesis that these demographic groups have fairly distinct interests from other groups. With the two topic groups that seem to involve more than one demographic group, the user can further understand how the demographic groups are intertwined though interactive analysis.

User Interaction. Hovering the mouse over one glyph that represents a particular demographic group highlights all other redditors belonging to the same demographic group. Such interaction allows easy discovery of whether redditors in the same group have cohesive or diverse interests. Such interaction also permits rapid analysis of clusters that seem to involve more than one group. Figure 1B shows two clusters with each including two different demographic groups. Such pattern suggests that the two demographic groups share similar topic interests based on their reddit posts.

The cluster view is coordinated with other views. Hovering over a certain demographic group in the cluster view will lead to highlighting of the corresponding demographic group in the ParSets and the corresponding topics. Conversely, when highlighting a particular demographic group or a topic in the ParSets+word cloud view, the corresponding group will be highlighted in the cluster view.

5.3 Feature ranking view

To address T3, namely allowing users to analyze the connection between demographic information and user generated content using features beyond topics, and to represent the predictive power of various features, we provide a feature ranking view (figure 5). In the tabular view, the features are aligned by row and the demographic variables are aligned by the column.

As introduced in section 4.1.3, the top table (with blue background) presents the contribution of different features in predicting the demographic variables. For instance, to predict gender as being male, linguistic features make the biggest contribution at 33.15%. Subreddit is the next best feature for such prediction based on user generated content. The background color of the cells is blue and the opacity is determined by the percentage displayed in each cell.[3]

[3]0% is likely due to insufficient data

Figure 6: Female, 17 or younger, less than high school degree and the corresponding topic of interest.

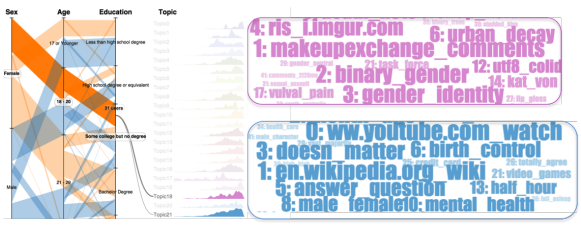


Figure 7: Female, 18-20, Some college but no degree and related topics.

The bottom table presents the cumulative accuracy of the predictive analysis for different demographic variables. Since the accuracy for each demographic variable in a column is analyzed in a cumulative fashion, we want to use the background encoding to reflect the accumulation. The background of the cells in the bottom table is a horizontal bar graph, with the length of each bar determined by the accuracy results. The contribution of the feature analysis view is that it enables the interactive analysis of the contribution of different features to demographic information prediction.

User Interaction. Since the linguistic feature contains 82 dimensions, with each dimension as an interpretable sub-feature, DemographicVis supports the expansion of the linguistic features to show more detailed ranking information. When expanded, the top 5 highest ranked linguistic features (figure 5 red rectangle) are presented. Hovering over a cell brings up the definition of the linguistic feature (figure 5 blue rectangle). Investigating Female and Male groups in our user pool, one can see the important linguistic features for female are anx (anxiety, e.g. worry, fearful, nervous), verb (common verbs), and cogmech (cognitive processes). Different set of features are highly ranked for male groups, including family, money, and religion. Such interactive analysis can potentially lead to significantly deeper understanding of how the different demographic groups talk and behave online.

6 Case Studies

In this section, we present case studies to illustrate how DemographicVis could help users compare and make sense of different demographic groups based on user-generated content from Reddit. Given that the majority of the redditors are young users, we take this opportunity to examine the interests of young crowds.

6.1 What are young ladies interested in?

We start by examining the group of "female, 17 or younger, less than high school degree". The individuals in this group are teenage girls attending high school or middle school. The most probable topic for describing their discussions is highlighted when hovering over the group, as shown in figure 6. The topic summarizes the young crowds' interests on reading Japanese graphic novel ("nurarihyon_mago") [4], discussing makeup related terms ("nail_polish")), and mentions of "panic_attack", which could be concerning since clicking on the term leads to posts that discuss mental illness.

Next we move on to selecting group "female, 18-20, some college but no degree". Individuals in this group are likely college students. As shown in figure 7, two topics are highlighted to describe the interests of this group. One can see both continuation and

[4]At first we thought this finding is due to the particular sample on Reddit. However, a study [23] based on 75,000 Facebook uses also found dominating interests in Japanese comics among young users.

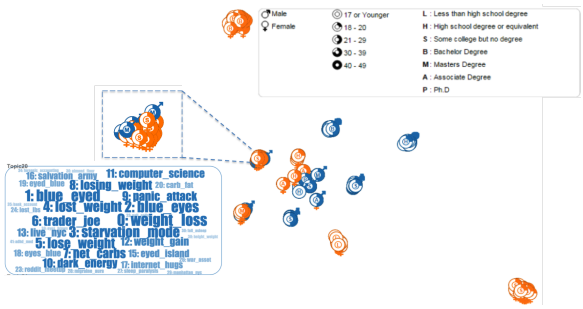


Figure 8: Cluster view that groups Reddit users based on their topic interests. One group with two demographic groups is enlarged and their shared topic is shown.

evolution in topic of interests compared to the younger generation. While continuing the discussion of makeup related terms "makeupexchange_comments" and "urban_decay"[5], young ladies within this demographic group also discussed gender identity and related issues: "male_female", "gender_identity", "birth_control".

6.2 Exploring diversity and overlap in interests among demographic groups

To identify whether there are overlaps between various demographic groups, we leverage the cluster view in DemographicVis. As shown in figure 8, reddit users are grouped based on their topic interests. Through a quick glance, one can find separable and cohesive clusters around the outer part of the view. For instance, the big orange cluster on the top (female, 21-29, bachelor degree) and the blue cluster on the bottom (male, 17 or younger, Less than high school degree) illustrate that the two groups have fairly distinct but cohesive interests. Hovering the mouse over a cluster will highlight the corresponding topics in the ParSets+Word Cloud view[6]. Interestingly, the clusters with different demographic groups denote groups that share similar interests. Here we pick a cluster (a zoomed in view of the cluster is shown in figure 8) with two demographic groups, namely "female, 21-29, some college but no degree" and "male, 30-39, masters degree". Hovering the mouse over the two groups leads to the identification of the common topic shared by the two groups, as shown in figure 8. Through a quick examination of the posts and subreddit where the two groups of users published their posts, we can then observe that the two groups are both interested in "weight_loss, lose_weight".

7 User Study

In this section, we present a user study to evaluate efficacy and usability of DemographicVis by comparing it with the SAS Text Miner [21]. The SAS Text Miner is one of the advanced analytics products developed by SAS that aims to extract insights from unstructured text. We consider SAS Text Miner as the best candidate for comparison because it integrates analysis and visualization capabilities for textual data.

7.1 Experiment tasks and apparatus

We designed 3 tasks for participants to perform with both DemographicVis and the SAS Text Miner. The main goal is to ask the participants to investigate the demographic groups and identify the connection between the demographic groups and the topic features. When designing the tasks, we wanted to test users' understanding of predictive analysis based on various features. However, we could not find functions in the SAS Text Miner that support this task. The 3 tasks we settled on are:

[5] A cosmetic brand

[6] The mixed glyphs in the center of the figure mainly contain the demographic groups that do not have sufficient numbers of users; as a result, it is difficult to model their interests.

Task1 : Identify 3 most frequently discussed topics by each demographic groups.

Task2 : For the 3 topics identified in Task1, find the corresponding demographic groups that mainly discuss these topics.

Task3 : Given 3 randomly assigned topics, rate the interpretability of the topics.

Task1 requires users to pick topics that are discussed most often by volume; both DemograhicVis and Text Miner provide functions to complete such a task but with different visual representations (topic stream vs. pie charts). Task 2 is to identify the demographic groups that discuss the 3 topics picked during the first task, while task 3 focuses on the interpretability of the topics extracted from both systems. In DemographicVis, the users can also access the Reddit posts (by clicking on terms in the topics) to aid the interpretation of the topics.

30 users participated in the user study, 19 males and 11 females (17 Ph.D. students, 5 masters, and 8 undergraduates). The age of the participants ranged from 18 to 40 (M=26). Participants were first asked to fill out a pre-study questionnaire regarding their demographic and experience with text analytics visualizations and Reddit use. We found that not many participants frequent Reddit.com (M=1.9 on a scale of 1 to 7). Many participants rated that they are pretty familiar with visualizations (M=3.5 on a scale of 1 to 7), such as word cloud and tree map. Out of the 30 participants, 5 participants answered that they were familiar with the SAS tool and 7 answered that they are familiar with text summarization methods.

We prepared two training videos on DemographicVis and SAS Text Miner. Prior to starting the study with each interface, the participants watched a 3 minute training video that demonstrates how to use the interface for completing the tasks. The two interface conditions were presented in counter-balanced order across all participants. After a participant finished the user study with both interface conditions, she was asked to fill out a post-study questionnaire regarding subjective preferences on the interfaces as well as each interface's advantages and disadvantages. All participants successfully finished the user study within 60 minutes.

7.2 Results

In this section, we report the results from the user study. Subjective ratings (7-point Likert scale) on the two systems are reported. We also summarize the user feedback on the pros and cons of both systems. The subjective rating data were analyzed with Friedman's test with α=.05 level for significance. There is a huge debate ongoing in the socialbehavioral sciences over whether Likert scales should be treated as ordinal or interval. We choose to treat it as ordinal, therefore Friedman's test is used to analyze the results.

7.2.1 Accuracy rate

For Task1, the participants were asked to pick 3 most discussed topics. We found no statistically significant difference between DemographicVis and SAS Text Miner (M=0.93 vs. M=0.96). Both systems provide functions that enable such identification.

For Task2, we asked participants to find the corresponding demographic groups for the top 3 topics. We calculate error rate as the number of incorrect answers. The result of an one-way ANOVA shows a significant main effect on accuracy rate of interface condition ($F(1,29)$=7.760, p=.009, η_p^2=.211). When using DemographicVis, participants exhibit a higher accuracy rate (M=2.83, SD=0.59) than SAS Text Miner (M=2.17, SD=1.15). Based on the user feedback, the reason that DemographicVis outperformed the SAS Text Miner in this task is because of the interactions supported to help users easily connect topics with demographic groups. Although the SAS Text Miner provides similar functions, lack of direct manipulation on the visualization (pie charts) and coordination between multiple windows makes it difficult for users to identify the demographic groups.

For Task3, although DemographicVis and SAS Text Miner use different methods to extract topics, the participants rate the interpretability ($\chi^2(1)$=.182, p=.670) as comparable.

7.2.2 Learnability and Usability

Participants rated learnability (easy to learn) and usability (easy to use) for each interface after performing all tasks (on a 7-point Likert scale, 1:*very difficult* to 7:*very easy*). The results of Friedman's test show that there is a significant difference on a learnability rate ($\chi^2(1)$=6.545, p=.011). Median (IQR) learnability rates for DemographicVis and Text Miner are 5.5 (4.75 to 6) and 5 (3 to 6). In addition, there is a significant difference on a usability rate ($\chi^2(1)$=8.048, p=.005). Median (IQR) usability rates for DemographicVis and Text Miner are 5 (4 to 6) and 5 (3.75 to 5.25). Overall, participants rated that DemographicVis is easier to learn and use than the Text Miner to accomplish the designed tasks.

7.2.3 Subjective Preference

When asked which system one prefers for accomplishing Task1, 20 out of 30 preferred DemographicVis, 7 preferred SAS Text Miner and 3 answered both. For Task2, 24 out of 30 preferred DemographicVis, 5 preferred Text Miner and 1 answered no preference. For Task3, 21 prefer DemographicVis while 6 preferred Text Miner, 2 answered both are same and 1 answered no preference. In terms of overall preference, 25 out of 30 answered that they prefer DemographicVis and 5 answered they prefer SAS Text Miner. From the open-ended comments, we see many participants commented on features provided by DemographicVis that show correlations between topics and demographic groups including "Different views were synchronized and responsive", and "It was easier to detect the correlation of topics to groups in DemographicVis". In contrast, many commented on Text Miner's lack of view coordination "need to open extra windows", and lesser visualization quality"the nested pie charts are confusing".

8 Discussion and Conclusion

Throughout the design process, we noted that there are aspects we'd like to continue to improve in both data collection and analysis.

First, to be able to make substantial claims on findings regarding the interests and online behaviors of various demographic groups, we will need to collect a much larger sample. In practice, this is difficult to achieve on Reddit.com alone since the only place one is allowed to post surveys is the r/SampleSize subreddit. We plan to conduct similar surveys on other social media platforms such as Twitter. It will be interesting to conduct comparative analysis on what the demographic group publish on different social media sites.

Second, we would like to improve the feature analysis process to achieve better predictive results. Having a larger sample could help, but more features would also be added to boost the performance of the the predictive results. Some features may be platform specific. For example, in our experiment, subreddit turns out to be a good feature for predictive analysis. Other general features such as how often a user posts (indicating how active she is) or how many different subreddits or topic groups the user posts in may also contribute to the overall predictive analysis. In terms of using the topic features for predictive analysis, we can experiment with different topic models to see which one may yield better results. We did experiment on generating different number of topics with our NMF-based topic model, and found that the number of topics does not affect the predictive results and the contribution of the topic features.

9 Conclusion

We introduce DemographicVis, a visual analytics system that aims to support interactive analysis of demographic information based on user-generated contents. DemographicVis visualizes features that are extracted to either describe or predict demographic factors, and enables the exploration of demographic information in a transparent manner. Results from our comparative evaluation shows that DemographicVis is quantitatively competitive and subjectively preferred compared to the SAS Text Miner.

Acknowledgements

The work is supported in part by the Army Research Office under contract number W911NF-13-1-0083 and the U.S. Department of Homeland Security's VACCINE Center under award no. 2009-ST-061-CI0002.

References

[1] N. Aletras and M. Stevenson. Evaluating topic coherence using distributional semantics. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013)–Long Papers*, pages 13–22, 2013.

[2] E. Alexander, J. Kohlmann, R. Valenza, M. Witmore, and M. Gleicher. Serendip: Topic model-driven visual exploration of text corpora. In *Visual Analytics Science and Technology (VAST), 2014 IEEE Conference on*, pages 173–182, Oct 2014.

[3] S. Argamon, M. Koppel, J. W. Pennebaker, and J. Schler. Mining the blogosphere: Age, gender and the varieties of self-expression. *First Monday*, 12(9), 2007.

[4] B. Bi, M. Shokouhi, M. Kosinski, and T. Graepel. Inferring the demographics of search users: Social data meets search queries. In *Proceedings of the 22nd international conference on World Wide Web*, pages 131–140. International World Wide Web Conferences Steering Committee, 2013.

[5] J. D. Burger, J. Henderson, G. Kim, and G. Zarrella. Discriminating gender on twitter. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1301–1309, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[6] J. Choo, C. Lee, C. Reddy, and H. Park. Utopian: User-driven topic modeling based on interactive nonnegative matrix factorization. *Visualization and Computer Graphics, IEEE Transactions on*, 19(12):1992–2001, Dec 2013.

[7] C. Chung and J. W. Pennebaker. The psychological functions of function words. *Social communication*, pages 343–359, 2007.

[8] W. Cui, S. Liu, Z. Wu, and H. Wei. How hierarchical topics evolve in large text corpora. *Visualization and Computer Graphics, IEEE Transactions on*, 20(12):2281–2290, Dec 2014.

[9] W. Dou, X. Wang, R. Chang, and W. Ribarsky. Paralleltopics: A probabilistic approach to exploring document collections. In *Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on*, pages 231–240, Oct 2011.

[10] D. Elliott. Crowd gazing - understanding demographic forces can help us better prepare for the problems caused by the world's rapidly expanding population. Accessed: 2015-03-30.

[11] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin. The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer*, 27(2):83–85, 2005.

[12] F. Iacobelli, A. J. Gill, S. Nowson, and J. Oberlander. Large scale personality classification of bloggers. In *Affective Computing and Intelligent Interaction*, pages 568–577. Springer, 2011.

[13] R. Kosara, F. Bendix, and H. Hauser. Parallel sets: Interactive exploration and visual analysis of categorical data. *Visualization and Computer Graphics, IEEE Transactions on*, 12(4):558–568, 2006.

[14] N. Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine learning*, 2(4):285–318, 1988.

[15] Z. Ma, W. Dou, X. Wang, and S. Akella. Tag-latent dirichlet allocation: Understanding hashtags and their relationships. In *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on*, volume 1, pages 260–267. IEEE, 2013.

[16] A. Mukherjee and B. Liu. Improving gender classification of blog authors. In *Proceedings of the 2010 conference on Empirical Meth-*

ods in natural Language Processing, pages 207–217. Association for Computational Linguistics, 2010.

[17] M. L. Newman, C. J. Groom, L. D. Handelman, and J. W. Pennebaker. Gender differences in language use: An analysis of 14,000 text samples. *Discourse Processes*, 45(3):211–236, 2008.

[18] J. W. Pennebaker, M. R. Mehl, and K. G. Niederhoffer. Psychological aspects of natural language use: Our words, our selves. *Annual review of psychology*, 54(1):547–577, 2003.

[19] J. W. Pennebaker and L. D. Stone. Words of wisdom: language use over the life span. *Journal of personality and social psychology*, 85(2):291, 2003.

[20] D. Rao, D. Yarowsky, A. Shreevats, and M. Gupta. Classifying latent user attributes in twitter. In *Proceedings of the 2nd international workshop on Search and mining user-generated contents*, pages 37–44. ACM, 2010.

[21] SAS. Sas text miner. `http://www.sas.com/textminer`. Accessed: 2015-03-30.

[22] J. Schler, M. Koppel, S. Argamon, and J. W. Pennebaker. Effects of age and gender on blogging. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, volume 6, pages 199–205, 2006.

[23] H. A. Schwartz, J. C. Eichstaedt, M. L. Kern, L. Dziurzynski, S. M. Ramones, M. Agrawal, A. Shah, M. Kosinski, D. Stillwell, M. E. Seligman, et al. Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one*, 8(9):e73791, 2013.

[24] S. Sheng, M. Holbrook, P. Kumaraguru, L. F. Cranor, and J. Downs. Who falls for phish?: a demographic analysis of phishing susceptibility and effectiveness of interventions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 373–382. ACM, 2010.

[25] L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008.

[26] Y.-C. Wang, M. Burke, and R. E. Kraut. Gender, topic, and audience response: an analysis of user-generated content on facebook. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 31–34. ACM, 2013.

[27] Y. Wu, S. Liu, K. Yan, M. Liu, and F. Wu. Opinionflow: Visual analysis of opinion diffusion on social media. *Visualization and Computer Graphics, IEEE Transactions on*, 20(12):1763–1772, Dec 2014.

Ensemble Visual Analysis Architecture with High Mobility for Large-Scale Critical Infrastructure Simulations

Todd Eaglin and Xiaoyu Wang and William Ribarsky and William Tolone
Charlotte Visualization Center, Department of Computer Science The University of North Carolina at Charlotte, Charlotte, NC 28269, USA

ABSTRACT

Nowhere is the need to understand large heterogeneous datasets more important than in disaster monitoring and emergency response, where critical decisions have to be made in a timely fashion and the discovery of important events requires an understanding of a collection of complex simulations. To gain enough insights for actionable knowledge, the development of models and analysis of modeling results usually requires that models be run many times so that all possibilities can be covered. Central to the goal of our research is, therefore, the use of ensemble visualization of a large scale simulation space to appropriately aid decision makers in reasoning about infrastructure behaviors and vulnerabilities in support of critical infrastructure analysis. This requires the bringing together of computing-driven simulation results with the human decision-making process via interactive visual analysis. We have developed a general critical infrastructure simulation and analysis system for situationally aware emergency response during natural disasters. Our system demonstrates a scalable visual analytics infrastructure with mobile interface for analysis, visualization and interaction with large-scale simulation results in order to better understand their inherent structure and predictive capabilities. To generalize the mobile aspect, we introduce mobility as a design consideration for the system. The utility and efficacy of this research has been evaluated by domain practitioners and disaster response managers.

Keywords: Disaster Forecast, Critical Infrastructure Simulation, Visual Analytics, Mobile Interface

1. INTRODUCTION

Forecasting the destructive impact for a volatile hurricane on a network of vulnerable critical infrastructure network is a central challenge for emergency planners and responders in hurricane-prone areas. After witnessing the devastating destruction from Hurricane Sandy, decision makers in coastal US cities are on high-alert for threats to their critical infrastructures (e.g, power lines, food networks, shelters, etc.); they are requesting more robust simulation analyses to depict the potential impacts from another tropical storm.

Much prior research has focused on using simulations and predictive modeling to anticipate hurricane movement and suggest possible landfall and impact locations.[1–3] Due to the complexity and scalability of simulation runs, understanding these modeling efforts and their predictive capabilities from large collections of simulation results is challenging.[4] On the one hand, many of the traditional hurricane modeling approaches depend on a trial-and-error approach that is not always feasible for generating simulations consistently. While this type of approach is widely adopted, each simulation run requires analysts to fine-tune the parameters, which can be very time consuming and less methodological.[5]

On the other hand, a new simulation approach is based on the idea of data-farming, which prepares many possible simulation outcomes in bulk.[6] However, this approach is largely limited by the simulation models, for which very few modelers have sufficient computing resources available to do sensitivity studies, validation and verification, effectiveness analysis, and related necessary activities. Exacerbating this challenge is that the large amount of such simulation results is far outpacing decision makers capability to analyze and make use of them.

Further author information: (Send correspondence to Todd Eaglin)
Todd Eaglin: E-mail: teaglin@uncc.edu, Telephone: 1 704 687 7989
Xiaoyu Wang: E-mail: Xiaoyu.Wang@uncc.edu
William Ribarsky: E-mail: ribarsky@uncc.edu
William Tolone: E-mail: William.Tolone@uncc.edu

To meet the challenges of dealing with disaster forecast and preparation, automated simulation methods are essential. However, they are not sufficient. There must be human input and direction, specifically for the interpretation of results and in some cases for directing the simulations. It is important for the decision makers to gain enough actionable knowledge such that they can respond in a timely and correct manner to possible failures of crucial infrastructure.

A key design component for our system is *Ensemble Visualization* of a large simulation space generated as a result of the aforementioned data-farming simulation approach. An ensemble, in this case, is a high-dimensional collection of attributes aggregated from raw simulation results that are centered around a single feature (e.g., Power Station, Airport, or Hospital). Details of the attributes are illustrated in Figure 2.

Another key design consideration is *Mobility*, which, in our case, refers to mobile computing devices or environments that enable analysis in the field. It isnt limited to just personal devices (e.g., iPad), but more broadly includes moveable equipment in general. The demand for mobility has been demonstrated in in our previous police and evacuation exercises,[7] where our campus police employed a networked tablet, smart phones, and a mobile command center to provide situationally-aware support.

Central to the goal of our research is, therefore, the use of ensemble visualization of a large scale simulation space to facilitate decision makers reasoning about impact on critical infrastructures with mobility. It is in this spirit that we architected our scalable visual analytics system for analysis, visualization and interaction in order to better understand their inherent structure and forecast capabilities. Our system captures the interplay between cascading simulations of critical infrastructure, ensemble analysis, and a networked mobile visual analytics interface. It aims to balance both *human and computer intelligence* and provide situational awareness to decision makers in both planning for natural disaster response and taking direct action during a disaster.

1.1 Research Procedures and Aims

In line with our design goals, the first activities of our research focused on establishing a simulation space based on computationally traversing possible simulation permutations within a Cascading Infrastructure Simulation (CIS). Specifically, our system relies on massive amount of computer generated simulation results. Access to these results is crucial because a working system requires information about the variance of infrastructure impact within large geospatial areas, which are produced from the cascading model as input parameters are varied.

As detailed in Section 4.1, we have developed a CIS model that runs multiple hurricane paths to generate a raw simulation space with millions of infrastructure failures events. Currently, our simulation includes infrastructures such as electric power, telecommunications, water, and railroad transportation. Large amounts of failures events from these infrastructures provide the fundamentals for ensemble analysis, which enables a multi-scale exploration of the high-dimensional simulation space connecting the complementary insights from global and local analysis of the data. In addition, our simulation space is situational because it depends on the properties of the particular coupled infrastructure and also on the properties of the occurrence (which can vary) that brings stress to the critical infrastructure.

By using an ensemble of simulation runs, we enable users to sample the space of input conditions that is presumed to cover all possible starting conditions in a particular range, and then employ multiple models that provide greater or lesser fidelity in some aspect of the process. The resulting ensemble encompasses the range of plausible outcomes, and the variation within the ensemble members exposes information on simulation uncertainty and the sensitivity of input parameters.

To incorporate human-centered analysis, we further provide an interactive visual analysis interface for exploring the simulation space. The designed visual interface combines a variety of statistical visualization techniques to allow decision makers to quickly identify areas of interest, ask quantitative questions about the ensemble behavior, and explore the uncertainty associated with the data.

To meet the needs for mobility, we have encapsulated the interface into mobile devices, such as an iPad, with multiple coordinated visualizations to provide a cohesive view of the simulations and permit analysis at multiple scales. The coordinated visualizations and interactions are optimized for mobile display, following the design guidelines from Apple.[8] The complexity of the ensemble data will be mitigated by a flexible organization of the

information, and the coordination between views will permit users to focus on the formulation and evaluation of hypotheses. Specifically, our system is designed to help domain users address the follow questions:

- Which facility is more vulnerable than others?
- When will the facility be hit?
- How can responders in the field check buildings along a hurricane path in their local area?
- How is the hurricane's temporal development affecting the cascading infrastructure failure?

The rest of the paper is structured as following. We first characterize the research domain in Section 2, then describe related work in Ensemble Visualizations in Section 3. Details of our system will be introduced in Section 4. We provide our informal evaluation with disaster prevention planners in Section 5 and provide discussions in Section 6.

2. DOMAIN CHARACTERIZATION AND DESIGN CONSIDERATIONS

2.1 Ensemble Analysis of Simulation Space

The architecture we present in this paper aims to help monitor and adapt to infrastructure changes by addressing the hurdle brought by the sheer size and heterogeneity of the relevant critical infrastructure simulations for the state of North Carolina. Recent disasters have highlighted the great vulnerability of both coastal and inland communities, such as power outages and road blockages caused by Hurricane Irene and Hurricane Isabel (which surged inland through Charlotte). It is important to identify the weak links within the massive infrastructure networks before the next hurricane hits. Road maintenance plans, development policies, hazard mitigation, and emergency response plans depend upon an understanding of the scale and linked cascading effects from hurricane impacts on critical infrastructures (e.g.,links between power substations and water services).

Managing critical infrastructure simulations is a complex, multi-stage process. Understanding the impacts of stress on infrastructures requires an effective workflow that includes data acquisition, de-noising, analysis, visualization and interaction. Based on feedback from our emergency response collaborators, one major challenge within the CIS process is that available tools are inadequate for analyzing and managing such large simulation results. Based on a survey conducted by the National Oceanic and Atmospheric Administration (NOAA), 57 out of 198 coastal infrastructure managers showed the need for a decision-support systems that can help them monitor the potential impacts for each infrastructure from all simulation results.[9]

Our ensemble visualization is, therefore, setup to help understand how changes in the hurricane course and properties alter the stability of infrastructures. Our emphasis is on providing the ability to effectively and (semi-)automatically depict temporal and geospatial changes of coastal infrastructure caused by both historical and simulated natural disasters. Furthermore, we aim to enable users to view and interact with the simulations datasets, as well as the analysis results, in an unencumbered and intuitive fashion.

2.2 Interactive Visual Analytics Interface with Mobility

Mobility is another important aspect in our research effort. Through prior work in evacuation exercises, we have first-hand experience with campus police and DHS emergency planners to design and deliver networked mobile visualizations system using an iPhone.[7] Specifically,the campus police chief and his force requires on-the-go analysis while events are occurring. The police chief stressed the benefits to have a mobile app on his iPad that he could use in conjunction with the setup he already employs in the mobile command center. This would greatly improve the police response time and keep them constantly connected while in the field under the coordination of the chief.

Other portable devices like laptops are not as mobile as a tablet and are cumbersome for the first responders as they move around on foot. A tablet is easy to carry around and allows the responder to access what he needs at any time and place. In this regard, the chief even went further and claimed the analysis system with mobility as an invaluable tool to integrate into his current methods.

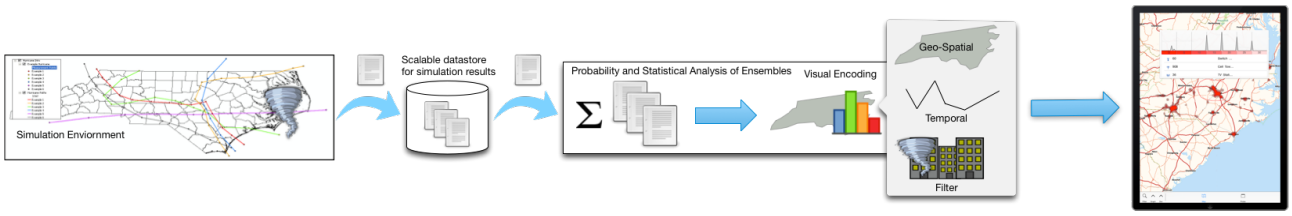


Figure 1. System Architecture of our Ensemble Vis.

During the first two years of our collaborations with responders and planners, we explored some of the simulation and data analytics challenges, and considered it critical to identify the most vulnerable infrastructure before a disastrous event, so that natural disaster impacts can be planned for and mitigated. Motivated by the above domain characteristics, our partnership, therefore, focuses on research on ensemble visual analytics over massive amount of infrastructure simulation results, with the goal of providing mobile visual interfaces that support analytic workflows to yield accurate and actionable results.

3. RELATED WORK

Our work is built upon the concept of Ensemble Analysis. A specific research area that deals with ensemble data sets is climate and weather data visualization. An ensemble dataset is defined as a collection of simulation features that are generated by computational simulations of one or more state variables across space and time.[10] As temporal analysis assumes an ordered sequence of functions, a slightly different analysis comes into play when no ordering is imposed on a set of functions. This type of data often arises not from time-series data, but rather from an ensemble of simulations. For example, in our targeted critical infrastructure simulation, stochasticity or randomness within the simulation setup may results in different observations following different runs. Such varying observations form an ensemble of functions.

The variation among the ensembles arises from the use of different numerical models, input conditions, and parameters. The complex nature of ensemble data sets leads to numerous possible approaches to visualization. Multivariate correlation in the spatial domain is a common approach for reducing the complexity of the task of data understanding, as is reducing the data to a hierarchical form which is conducive to 2D plots.[11] Büerger and Hauser[12] present an overview of techniques for multivariate data and Buja et al.[13] discuss a taxonomy of interaction with high-dimensional data.

One important challenge is to understand how the stochasticity or randomness within the simulation design impacts the infrastructure analysis of multiple runs. This could be considered as an uncertainty analysis that explores the relationship between input parameters and the output.[14,15] One strategy is to derive a statistical description of these ensemble structures: for example, by summarizing each simulation runs and then describing the statistical distribution of the entire ensemble simulation space. There has been some recent efforts in conducting statistical ensemble research, such as computing certain statistics on topological summaries .[16,17]

Ensemble analytics and visualization is a new and rapidly growing field where researchers in statistics and computational topology have just begun investigations, and a wealth of questions remain open. Software systems, such as SimEnvVis[18] and Vis5D,[19] are designed to handle atmospheric data formats and include 2D geographical maps with color maps and contours, as well as more sophisticated techniques such as iso-surfacing, volume rendering, and flow visualization. The Noodles system[20] provides the aforementioned capabilities and adds uncertainty contours and glyphs. In addition, Ensemble-Vis,[21] uses meteorological data to provide a visual exploration of short-range weather forecasting. It adapts visualization methods common to the domain of meteorology; it also adds indications of uncertainty and enables the user to drill-down to ensemble data.

Given the need to integrate human knowledge in the analysis process, our work also relies on ensemble feature integration. Here, we focus on the combination of interactive visualization and feature analysis that has resulted in a set of feature integration and exploration techniques for analyzing multi-dimensional simulation spaces. Prior work in this area can be grouped into three categories, two-variate visualization, multivariate visualization, and analysis animation,[22] depending on the nature of the simulation data. The utility of these

techniques has been demonstrated in the analysis fields of volumetric data,[23] terrain change detections,[24] and medical imaging.[25] However, these traditional feature integration techniques no longer scale with the increasing size and complexities of the simulation datasets. Our research aims to address several challenges in this research area including feature selection and comparison, interactive exploration and semi-automated annotation.

By integrating knowledge gained from our previous work done for a mobile crowd sourcing application to model 3D buildings,[26] we established our networked visual analytics architecture to build a better way to handle such mobility requirements while supporting data traffic, user interaction, and visualizations. Multiple linked views of data relieve the need to present all data of interest in a single window.[27] Such approaches let the user interactively select regions of interest and reflect those selections in all related mobile views. The resulting mobile interface organizes a collection of views to provide complex investigation of the data under user control.

4. SYSTEM COMPONENTS AND IMPLEMENTATION

The focus of our system design is to aid domain experts and planners in analyzing millions of simulation results in order to identify the vulnerabilities and resiliency of a critical infrastructure. As shown in Figure 1, our networked visual analytics system comprises three integrated components: A cascading CIS model connected to a shared Oracle database that generates the raw simulation space (Section 4.1); an ensemble analysis module that handles statistical computation of ensemble structures and provides an optimization and data retrieval API for the visual analysis module (Section 4.2); and a mobile visual analysis interface that encodes the ensemble results into multiple coordinated visualizations for conducting field examinations and preparations (Section 4.4). The following sections describe each of the components in greater detail.

4.1 CIS: Generating the Raw Simulation Space

To capture the complex, multidimensional, and dynamic effects from a hurricane, we utilize a CIS simulation model [blinded for review] that takes into account the interrelationships among critical infrastructures. Built within a rule-based framework for integrating multiple infrastructure components at a high level, our approach allows the user to create a model which represents the users assumptions about the world.This results in a dependency/interdependency ontology. Thus, for example, a breakdown of a power substation would immediately cascade to power loss at points on its distribution network. If a school were a node in the distribution network, it would be switched to backup power that, after a given time, would also shut down.

In addition, our model handles the concept of output requirements which allows the user to specify which inputs are necessary to produce the specified output from the feature. This is especially useful to help users gain focus on the commodities that matters most to their planning or response amid the noise of the overall simulation. For example, a power-grid specialist can select a subset of inputs that only applies to the multi-linked power station within his jurisdiction and run partial simulation around it.

4.1.1 Model Inputs and Simulation Concepts

Our CIS model consists of a wide range of data inputs, such as features, commodities, relationships, networks, and latencies. To keep the model results as realistic as possible, we conducted, over a 6 month period, a thorough collection of infrastructure datasets within the Carolinas.

Specifically, *features* are the physical infrastructures involved in the model such as communication features (e.g., cell towers), electric network features (e.g., power substations), and services features (e.g., hospitals and restaurants). In our current implementation there are 48 infrastructure including public schools and shelters.

The majority of this data comes from FEMA HAZUS data disks from 2005.[28] The cellular towers, communication facility and TV stations were provided by the FCC.[29] We further generated the switch control data for the communication network based on ESRI's random point generator.[30] We have acquired or generated a rich set of electrical network features. Not only did we acquired a set of HAZUS data points for power generation facilities, we had also generated a human-annotated network features by employing undergraduates to digitize the transmission power lines and the substations from orthophoto satellite images.[31] Moreover, the service feature data is largely collected through out domestic and international collaborators.[32] The food features where provided by a project partner.

name	Meaning
id	Feature ID stored in DB
Object_Class_Name	Commodity Type (e.g. Fire station)
Simulation_ID	Index Number for Simulation
Commodity_Name	Type of the Commodity
Indirect_Cost	Is the commodity disable indirectly?
Direct_Cost	Is the commodity disable directly?
Relative_Time_In_Hours	Time of Disablement
Number_Hours_Disabled	How long is the commodity disable during the simulation
Cause_Event_ID	Disable Event ID
COA_ID	Cause of Action ID
Model_ID	Which Hurricane Path Model
Network	Which Infrastructure Network (e.g.,Power or Transportation)

Figure 2. Detailed dimension information for each Ensemble.

Networks are, therefore, systems of infrastructure features that all share similar connection properties. The electric grid, road, rail, and pipeline system are all modeled as different networks with distinct properties that best reflect their actual applications.

To simulate the cascading effect, our model takes in the concept of a *commodity*, which is a tangible or conceptual good flowing between selection sets. In total, we have collected a total of 15 commodities such as electric, water, natural gas, manufactured goods, personal relationships, and wealth.

4.1.2 Cascading Relationship for Model Accuracy

This collection of networks and commodities gives us a good foundation for performing realistic cascading critical infrastructure simulations. Within each network, a commodity flows between two features through the creation of a relationship. The relationship establishes the provider and consumer of the commodity, the criticality of the good transfer and the method for establishing relationships. In our current model, we have pre-defined over 80 relationships that aims to make the model results as close to reality as possible.

Thus, according to the definitions above, the interlaced critical infrastructures are captured in a set of networks with each node having a set of properties according to its category with the edges providing a dependency rule according to the category and state of the two connected nodes. For example, a breakdown of a power substation would immediately cascade to power loss at points on its distribution network. If a hospital were a node in the distribution network, it would be switched to backup power that, after a given time, would also shut down. In this way, our CIS model takes cascading events into account.

To enable users to hypothesize different cascading conditions, we have built four methods to create a relationship: explicit, nearest neighbor with/without redundancy, and spatial. The users can directly specify the direct link between two features through an *explicit* relationship between the origin and destination. While the explicit method specifies a strict point-to-point relationship, the users can also use the *nearest neighbor method* to create connections between the origin and destination features that are within a specified distance. The *redundancy option* is built to further narrow down the connectivity between features. A relationship with redundancy means that the destination node is connected to all sources within the distance (as in the case of cell towers), while without redundancy means the destination node is only connected to the closest provider (as in the case of water

pipelines). For example a hospital could connect to any sewer pipe within 1000 meters but it only creates a relationship with the sewer pipe closest to it. Finally, a *spatial relationship* is an instance where the origin and destination share a relationship based on proximity of their location to the nearby features. In our model all electric relationships are based on spatial relationships.

4.1.3 Simulation Runs and Results

Our simulation runs with on a set of Courses of Actions (CoAs), which are a pre-determined encapsulation of the above factors to reflect the state changes for particular critical features in the storm path. Through a COA, a user can specify events that occur during the simulation and inform the CIS of What (selection set instance) undergoes a state change (e.g., intact or disrupted) and When (relative time into simulation) this occurs. The user can make the COAs permanent or temporary. A permanent event means that the feature will stay at the new state until another COA event changes the state. A building that has been physically destroyed is disabled and cannot be re-enabled even if all of the buildings inputs are available to it until the user specifies that the building has physically been rebuilt.

Latency is an important part of the CoA, which is designed to represent the delays before a selection set undergoes a state change. Latencies can account for backup electrical power or, in the case of food stores, the time periods until food supplies are exhausted. For example, a hospital can have a disablement latency of 24 hours for blood supply but a 14 day electric disablement latency.

We further constructed six automatic model variations to extend the conditions that are simulated. This set of model variation allows the users to test their assumptions on distance and the importance of redundancy within the system. Specifically, the six model variations only represent changes to these relationships but not to any other aspect of the model. The first model in each set is the base model which represents the users best guess on structure and type of relationship between the selection sets. The second model converts all nearest neighbor without redundancy (NN) and converts them to nearest neighbor with redundancy (NNR). The third model doubles all relationship distances. The fourth model takes those new distances and converts any NN to NNR. The fifth model takes the distances in the base model and halves them. The six model takes those new distances and converts the NN to NNR. Note, NNR is only applicable to where redundancy makes sense (e.g., not water pipes).

In order to cover the simulation space, we run a large number of simulations with different hurricane paths and different intensities and spreads. In total, we have run the six models across 12 different hurricane paths, including six historic hurricanes paths (see Figures 3) and six notional paths. Doing so allowed the users to gain a comprehensive view of what happens for different storm strengths or by changing our damage radii assumptions. The hurricane paths were created together with our collaborators and give our simulation model a good coverage across the Carolinas. The different intensities and spreads give a comprehensive view of what can happen under different conditions.

With over 50,000 features, so far, we have completed 425 simulations over the 12 hurricane paths and created 14.6 million simulation events. As shown in Table 2, each event is a high-dimensional collection of fields that are centered around a single feature (e.g., Power Station, Airport, or Hospital). All the data is stored in an Oracle database for remote access and analysis.

4.2 Extracting Ensembles from the Simulation Space with Statistical Analysis

While our CIS methods are essential for creating a large simulation result space, this is not sufficient. There must be human input and analysis, specifically for the interpretation of results. It is important for the decision makers to gain enough insights to form actionable knowledge so that they can respond and react effectively to possible failures of crucial infrastructure. This is the primary reason for us to enable interactive ensemble visual analysis of the raw simulation space.

Our ensemble visual analysis is used to abstract and reduce the complex and vast amount of information for each infrastructure. This enables the decision maker to understand the full crisis in its context and to detect potential cascading effects. It further permits the users to select the paths and CoAs most likely to occur for the most likely range of conditions. This will in turn indicate the most likely parts of the infrastructure to be

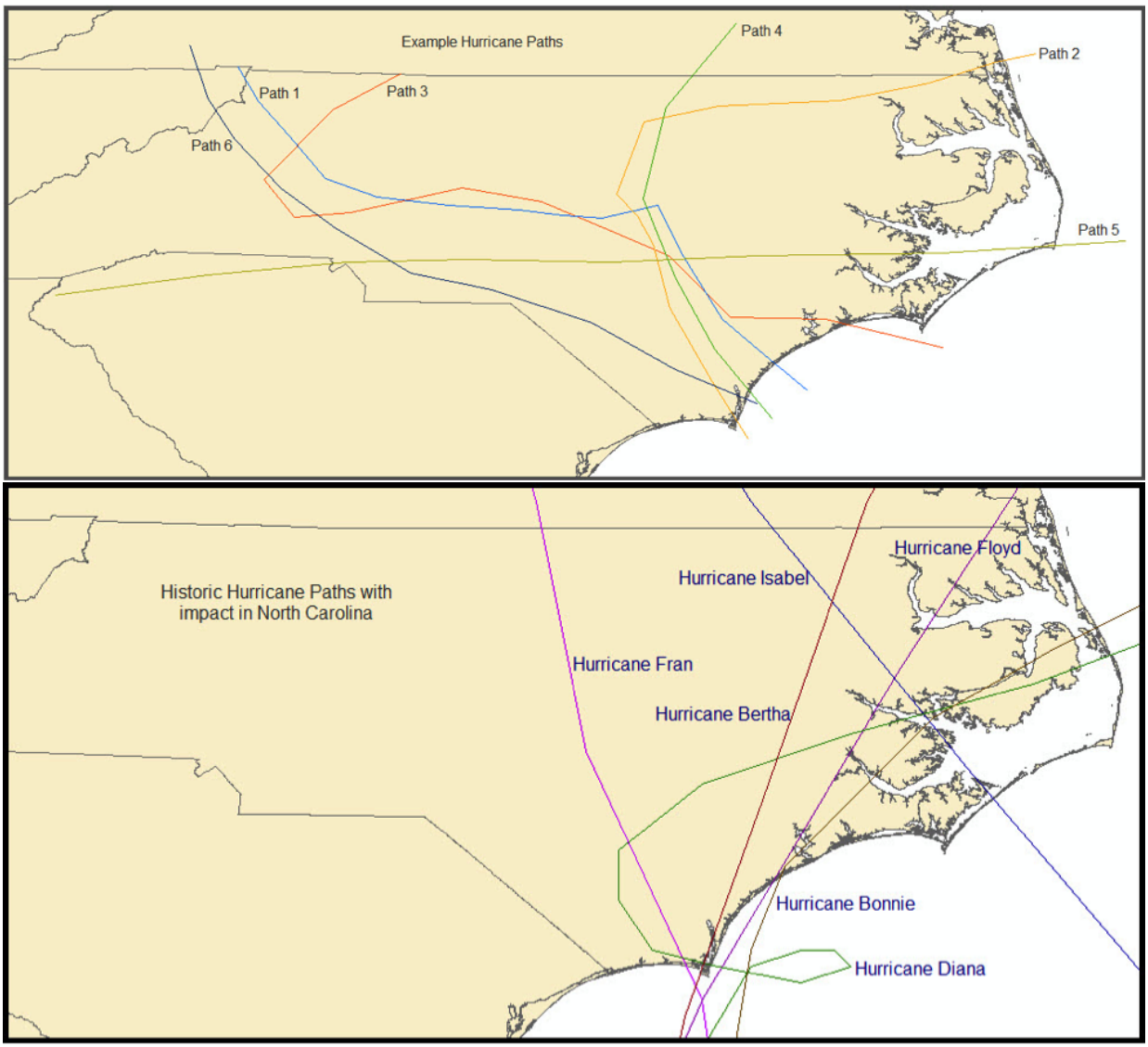


Figure 3. Two models simulated with our critical infrastructure modeling. Top is the six notional Hurricane Paths. Bottom is the six Historical Hurricane Paths, including Bertha, Bonnie, Diana, Floyd, Fran and Isabel.

disrupted and show the likely cascading effects. For example, this gives city emergency planners information on "how likely is my airport going to be hit by this storm and when will this occur?"

We define our ensemble as a high-dimension collection of fields aggregated from raw simulation results that are centered around a single feature. Specifically, we compute an ensemble as a collection of disablements across the entire simulation space for each specific infrastructure. The key value is a probabilistic outage for each infrastructure feature. This value represents the likelihood for a specific infrastructure feature to be disabled at any point during the simulation. We also compute the breakdown of causes for a specific infrastructure feature along with the minimum, median and maximum monetary cost imposed by it being disabled.

In addition, multivariate correlation[11] is used to capture the stochasticity or randomness within the simulation setup and reduce the complexity of the simulation event data. As a result, we are able to abstract and encapsulate the ensemble of every infrastructure. These infrastructures are represented by symbols, such as the glyphs (see Section 4.4.2) for power stations and cellular transmission stations.

Data Points	10	100	1,000	10,000	100,000
Cells	4mil	4mil	4mil	4mil	4mil
Server	0.011s	0.026s	0.079s	0.475s	7.07s
Mobile	0.28s	4.497s	34.807s	355.953s	~

Figure 4. This table shows the approximate time in seconds to perform kernel density estimation on varying data sizes in seconds. The spatial region was discretized into approximately 4 million cells. The mobile computation was performed on an iPad air using a CPU parallel implementation. The server computation was done using a Nvidia GeForce GTX 680MX using OpenCL.

4.3 Scalability Optimization for Mobile Analysis

Due to the inherent hardware limitations of commodity mobile devices, creating the needed mobile visual interface for millions of simulation events is both a computation and a visualization challenge. Where desktop computers can perform in-memory operations with large data, a commodity mobile device (e.g., iPad) is quite limited in its computing power. Therefore, a significant part of our research has been devoted to conducting scalable mobile optimization, which is heavily built around computing optimization and visualization scalability.

4.3.1 Offloading Ensemble Encoding to Computing Server

Our first optimization aims to accommodate the need to interactively visualize the entire ensemble space and its temporal changes. In this regard, we have employed a computing server that is designated to perform specific statistical calculations and data retrieval; these operations would otherwise be too time consuming on any mobile device.

Our computing server is connected to the database that contains all the raw simulation results. It handles data requests sent from the mobile device to the server, extracts the necessary data and finally computes the ensemble abstraction and encoding. We use a hash map for data transfer between the mobile device and the server. Each infrastructure has a unique key identifier. Using a hash map allows for quick lookups when doing additional computation on the mobile device.

In order to examine the temporal changes, our server performs statistical analysis on the ensemble through kernel density estimation (KDE).[33] This is done using GPU parallel computing and significantly offloads the demand for performing on-device computing. We use OpenCL to compute each cell of the KDE. This provides exponential speed up, but we have gone a step further since copying the computed data back from the GPU to main memory is usually more time consuming then the computation itself. To minimize data transfer between main memory and GPU memory we take advantage of being able to hand off the data computed from OpenCL to OpenGL for rendering. In the OpenCL kernel we can create geometry and then render it very quickly with OpenGL since it all exists in GPU memory. This allows us to not just create geometry very quickly but also pre-rendered bitmaps.

4.3.2 Utilizing Multi-core and the Graphics Engine for Mobile Rendering

Some processing must be done on the mobile device. These tasks usually involve data loading and retrieval as well as user interaction. Handling data can often be very time consuming but slowing the user interface is not conducive to our goals. When interacting with a mobile device any latency or slowness results in users becoming frustrated and hinders their productivity. Mobile devices often have multiple cores like their desktop counterparts so it is absolutely crucial to utilize any available resources. We utilize parallelism for all our web requests to create asynchronous data pulls. We also load all data from the CPU to the GPU through a background process. We purposely leave the user interface to its own separate process in order to maintain a consistent experience.

We also built a low-level graphics engine specifically for rendering the ensemble simulation space. It enables the device to render vast data sets very quickly. The graphics engine uses a scene graph style data structure for storing elements. It also utilizes graphics techniques like double buffering and constructing texture atlases on the fly for improved rendering.

Moreover, mapping systems included on most mobile devices are not capable of displaying lots of data very quickly and efficiently. They also are not portable between different platforms making cross-platform development difficult. We chose to develop our own mapping system in OpenGL making it very fast but also portable between platforms.

4.3.3 Remote Communications

Our mobile system is dependent upon the data pulled from the computing server. Any latency with this communication will affect the entire system. Due to this dependency we rely extensively on compression and aggressive caching to memory and disk. This limits potentially costly network requests and speeds up the mobile application.

We started with web services to query data on the fly, package the data and send it to the device. This method worked well with moderate datasets;as the data grew, however, the query time and packaging time began to slow down greatly. The response time had reached a certain threshold and at that point the application became unusable.

We therefore created a persistent environment instead of solely relying on a temporary service that queries the database each time and repackages it. This persistent environment caches the packaged data in main memory. Querying the database can be improved through minor tweaks with caching and other features, but what substantially slowed down the entire process was that each time the web service is called a new instance is spun up in main memory to load the queried data in order to package it into a form usable by the mobile device.

We utilize parallelism for all our REST (Representational state transfer) requests to create asynchronous data pulls and provide three possible return types for scalability. Each type is specified by the client based on its needs and capabilities. The first is a list of all points returned from the query stored in a compressed JSON file. Returning all the points allows the client to run additional operations that might not be computationally constrained. The second is a pre-computed kernel density estimation that is computed into indexed triangulated geometry and compressed into binary for transfer. This data type is for a client system that cannot calculate a spatial KDE in sufficient time, but is capable of rendering the triangulated result. The triangulated result is resolution independent in order to provide a higher level of detail for visual analysis. The last type is a pre-rendered bitmap of the kernel density estimation. This is for client systems incapable of handling raw data points or even triangulated geometry.

All these scalability optimizations setup a feasible platform for us to encode and visually represent the ensemble analysis into a mobile visual interface.

4.4 Ensemble Visualization with Mobility

As the first step in our ensemble visualization design, we followed prior visual analytics design studies (e.g.,[34,35]) and collaborated with first responders to understand their workflows and analytics needs. Based on our discussions, it became clear that a key aspect of our effort was to provide them with a visual analytics system that can accommodate their in-field analytics needs. Since, we are focusing on a system with high mobility like a tablet device our visualizations and analysis tools must be simple enough to use with touch gestures and smaller screens.

4.4.1 Geospatial-Temporal Overview for Ensemble Analysis

Customizable Geospatial View: Our simulation data set revolves around geospatial information. While the mobile OS incorporates mapping services they are not specifically designed for displaying large data sets or any complex 2D drawing. For this reason we incorporated a lightweight tile map server specifically for our needs. A tile map server[36] simply allows a client to fetch pre-rendered geo-spatial map tiles on the fly from a server and display them in an arranged fashion much like Google maps.

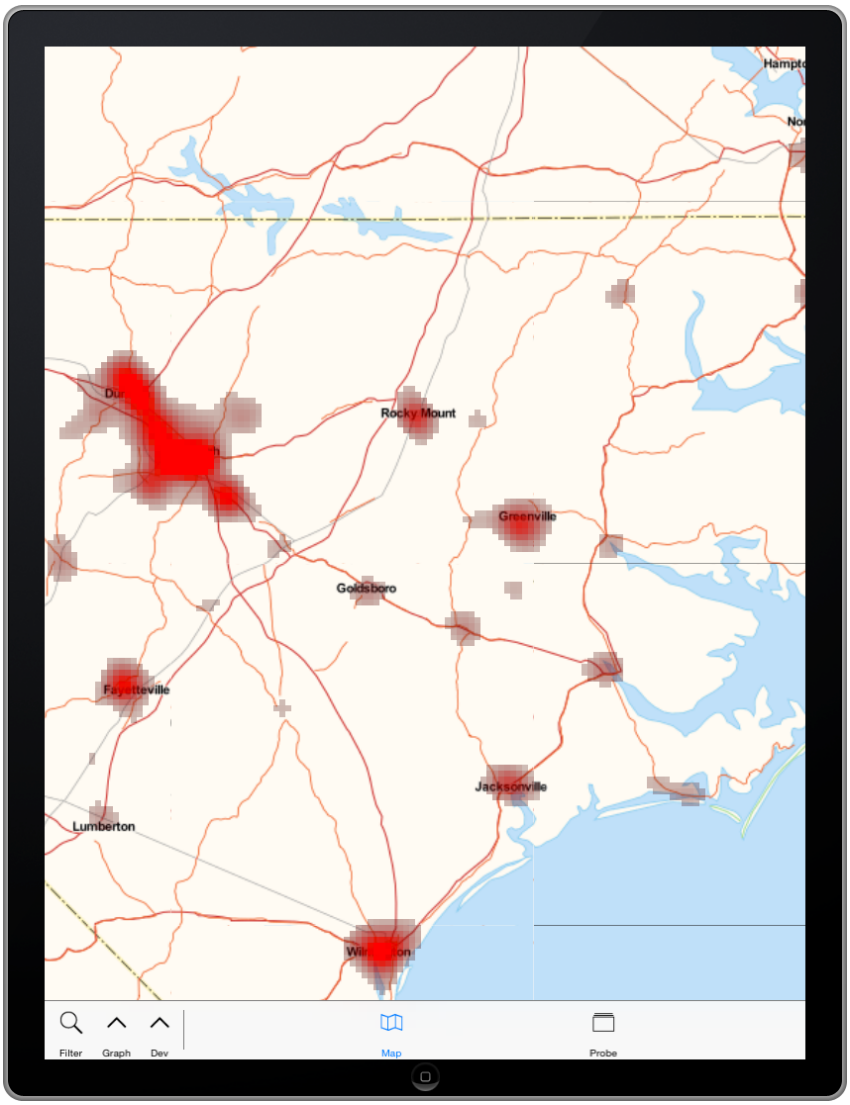


Figure 5. The Geospatial View of our mobile interface. The Kernel Density Estimation is represented in Red clusters.

Our customized tile map system, as shown in Figure 5 (A), bears a similarity to commercial mapping systems, but also provide additional functionalities that fits our visual analytics needs. Since the map tiles are stored in the cloud and the visualization components are implemented as layers that can be stacked onto the map, our mobile interface remains lightweight. This enables us to completely customize the maps and it also allows us to imbed certain features in the tile map textures to reduce the redundancy of rendering by having all the essential map details baked into pre-rendered tiles.

As shown in the video, we provide two different visualizations. The first is a time-based color coding of each individual infrastructure based on the average time it went out. This allows first responders to see when specific disabling events occurred during a hurricane. The second visualization is a Kernel Density map.[33] The intensity map shows the probability of outages for a particular region. This way users can quickly focus on regions with a high percentage of outages.

Temporal Visualization: Understanding temporal behavior of a disaster is another important aspect for emergency planners to isolate time critical infrastructure. We have developed an interactive temporal analysis view that enables the selection of specific time ranges to depict what infrastructure was disabled as the hurricane

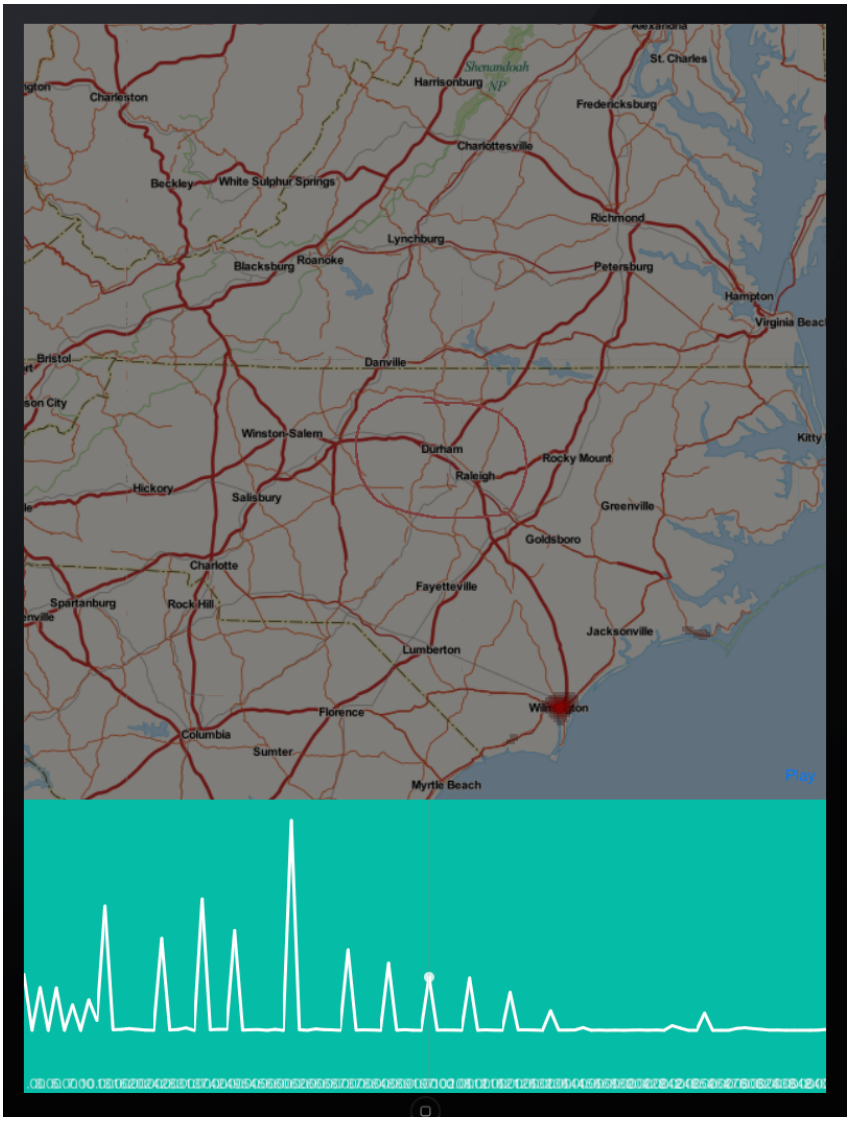


Figure 6. This image shows the interactive Temporal View. In this view, users can depict the peak time of the outages and select a range of time to further their investigations.

passed through. As shown in Figure 6, the user can filter the simulation space in time and further examine the outages leading up to a peak or the peak itself.

Our system also enables the users to examine the result of cascading outages and interactively analyze the complex cascading relationship between different infrastructures. This allows the users to visually depict the interdependency of infrastructures and understand the effect of one infrastructure on others. Through the interactions, our collaborators were able to observe hurricanes with two distinct impacts patterns, namely the immediate sparse outages following high peaks of outages as well as the latency outage pattern where an infrastructure feature (e.g., hospital) will stay on with its backup generator for a longer period of time before becoming disabled. These are all key temporal factors for evacuation and planning efforts.

Animation: We have further setup animations to help the users more effectively analyze the outage patterns. Previous research[37] has suggested animation plays an excellent role in revealing changes where cascading events take place. This gives an overview of the events that took place and hotspots during the simulation. However,

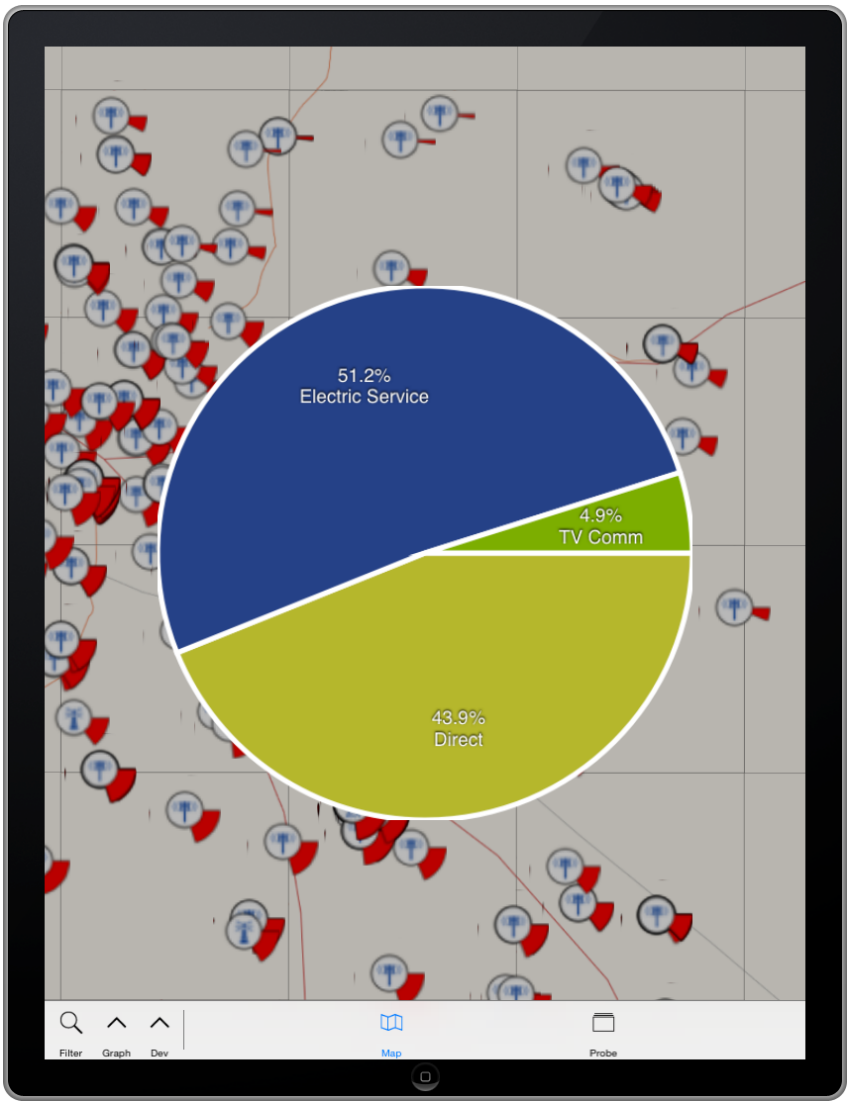


Figure 7. Detail view shows breakdown of impact on a feature based on Ensemble Analysis.

enabling animation posed a computational challenge given the 50,000 features and millions of events we have in our simulation space. As shown in Figure 4, computing each time segment for visualization exceeds the computation power of any existing commodity mobile device.

To address this challenge, we utilized streaming techniques that offload much of the computation to our server and streams the visualizations in quick succession to a mobile device. As shown in our video, we have created animation of all the simulation events over a thirty day period using spatial KDE.

In addition, the same three data types: data points, geometry, and pre-rendered bitmaps that are available for single queries are also available for animations. A client can request all the raw data points across the entire animation span or the triangulated geometry or pre-rendered bitmaps. This allows the animation to be flexible on a wide range of client needs and scalable even down to mobile devices.

4.4.2 Detailed View for Ensembles and Infrastructures

Each infrastructure in our simulation has a probabilistic chance that it will be disabled sometime across the entire simulation space. When the user zooms in close enough, we display a map glyph with the probability of each infrastructure as a pie chart, as shown in Figure 7. This allows the users to quickly see which infrastructure

Figure 8. Example of a subset of glyphs used to visualize specific critical infrastructure features. This figure shows symbols for cell tower, airport, emergency service, fire station, seaport, train station, hospital, power substation, and hotel.

features have a higher probability of being disabled. More importantly each feature might be disabled from multiple events. By selecting an individual infrastructure a secondary pie chart will be displayed that shows individual causes for disablement broken down by percentage. In this way a user can isolate the main cause for the infrastructure being disabled.

The glyphs for each infrastructure are created on demand based on the type of infrastructure and the summation of its potential outage. Figure 8 shows a few examples of different types of glyphs. The inner icon is associated with the type of infrastructure. This allows users to easily discern what specific infrastructure they are looking at secondly, the outer ring displays the probabilistic chance the infrastructure feature is disabled. A full circle would indicate a 100 percent chance of being disabled. This outer ring allows users to quickly identify what features have the greatest likelihood of being disabled.

4.4.3 Interactions

Probe Selection: Probing involves interactions that allow users to visualize a selected region in a free-form manner. As shown in Figure 9, a user can directly draw onto the map with his or her finger, drawing a bounding area around a region. This is an extremely important analysis tool because it allows the user to drive the analysis and focus on what is important to their geographic area. Anything within the bounding region will then be analyzed and returned to the user in a separate window.

When selected, a separate window will appear that initially shows the time distribution for all the features within the region. A user can quickly see any peaks in time in this window. Secondly, the user can then view the breakdown of what feature classes are in the selection and how many were disabled (Figure 7). This allows users to see what features were disabled, how many were disabled and the time distribution of those features, permitting a very fast analysis of regions of interest. It also allows users to pinpoint features that have a high probability of being disabled. Most importantly it allows a user to compare multiple regions for joint analysis.

Each probe selected by the user is copied and saved for additional viewing, as shown in Figure 9(Right). This way users can select multiple probes across many different areas and filters. Users can probe different time segments as well as different hurricane paths and commodities. Along with the selected features a screen shot of the region is attached to the data to give geospatial context to the region that was selected. All this information is presented in a secondary window where each probe is represented as a card with a map image of the selection and a time distribution of the events from the probe.

When the user selects a specific card the view transitions to a detailed breakdown of that specific probe. In this way users can return to further analyze individual events and time distributions from prior selections they have made.

Filtering: In addition, filtering is a key function that allows for the removal of unwanted features or events, as shown in Figure 10. In such a large simulation space, it is crucial for a user to be able to focus on specific commodities and hurricane paths. More particularly the combination of the two together allows for complex

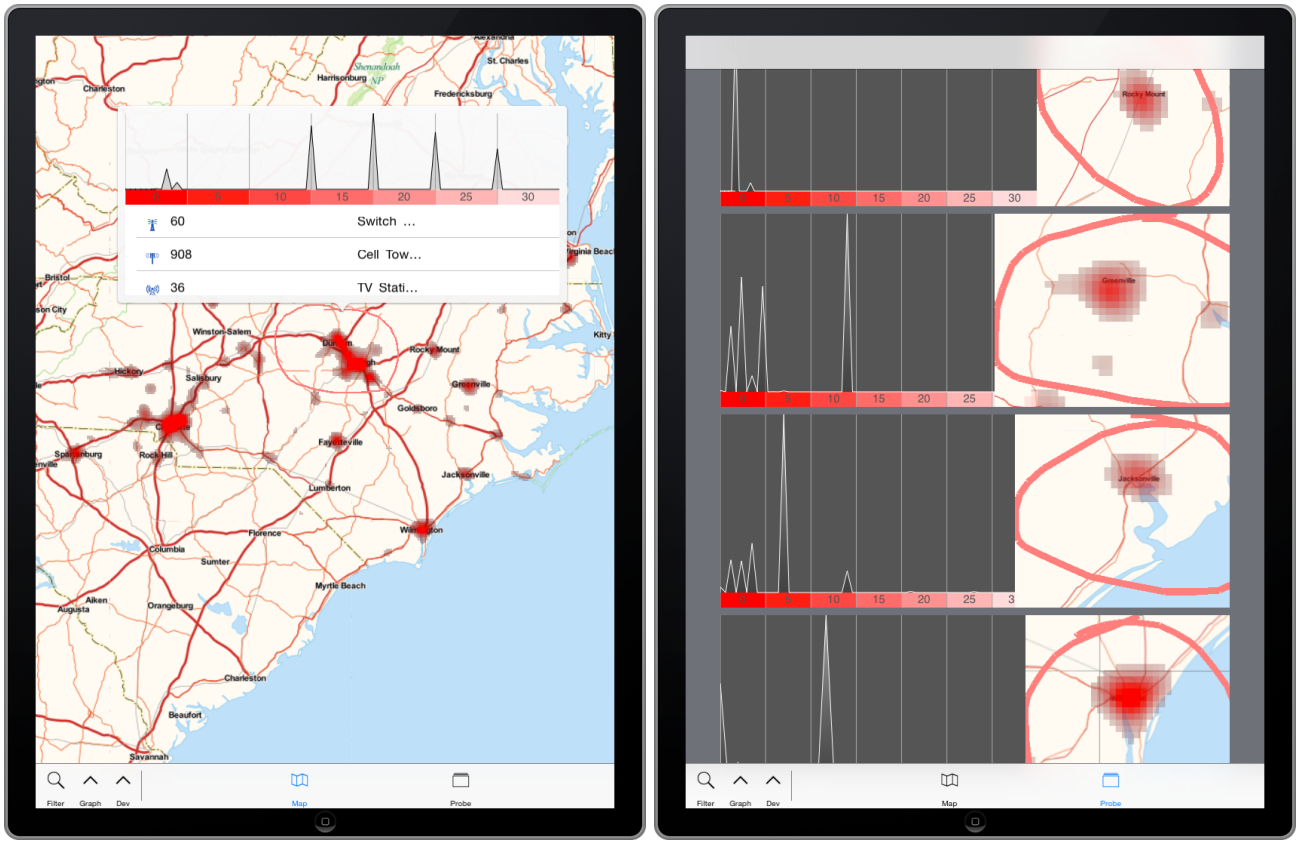


Figure 9. The probe selection view (Left) and a comparative view for all the probe selections (Right).

filters. This helps users navigate the simulation space by visualizing different parts of the simulation in a very easy manor.

The first type of filter is a commodity-based filter. Each commodity is separated into specific categories that share a common area. For example all the infrastructure features like ports, train stations, airports are separated into a transportation category. By having high-level categories, users can more easily pick what areas they would like to filter instead of navigating through lots of very specific features. Grouping similar features into categories also allows for easier comparison. We allow users to filter specifically on commodity types like transportation and electricity. This allows for quick comparison between different commodities.

The second type of filtering is hurricane path based. Our model has numerous hurricane paths that create a probabilistic outcome. Users can select specific hurricane paths to see the probabilistic effect of each impact. But, the real core of this feature is the filtering of multiple hurricane paths. By filtering on multiple hurricane paths pertinent to a region a user can get encompassing results for that region or quite specific results (e.g., what hurricane path is most likely to do damage to this particular infrastructure feature in my local emergency response area, and when).

4.4.4 View Coordination on Mobile Device

We took very careful consideration when designing view coordination on the mobile interface. As far as we know, no prior research has been done that studies multiple-coordinated views (MCV) for mobile. Due to the limited screen-space, our view coordination has to occur within the same display area, rather than in a spread out fashion as discussed in the original MCV guideline.[38]

We ultimately relied on the transparency of each view to make the view coordination apparent, as shown in the video for this paper. Since there is limited screen space on mobile devices the use of overlays and transparency

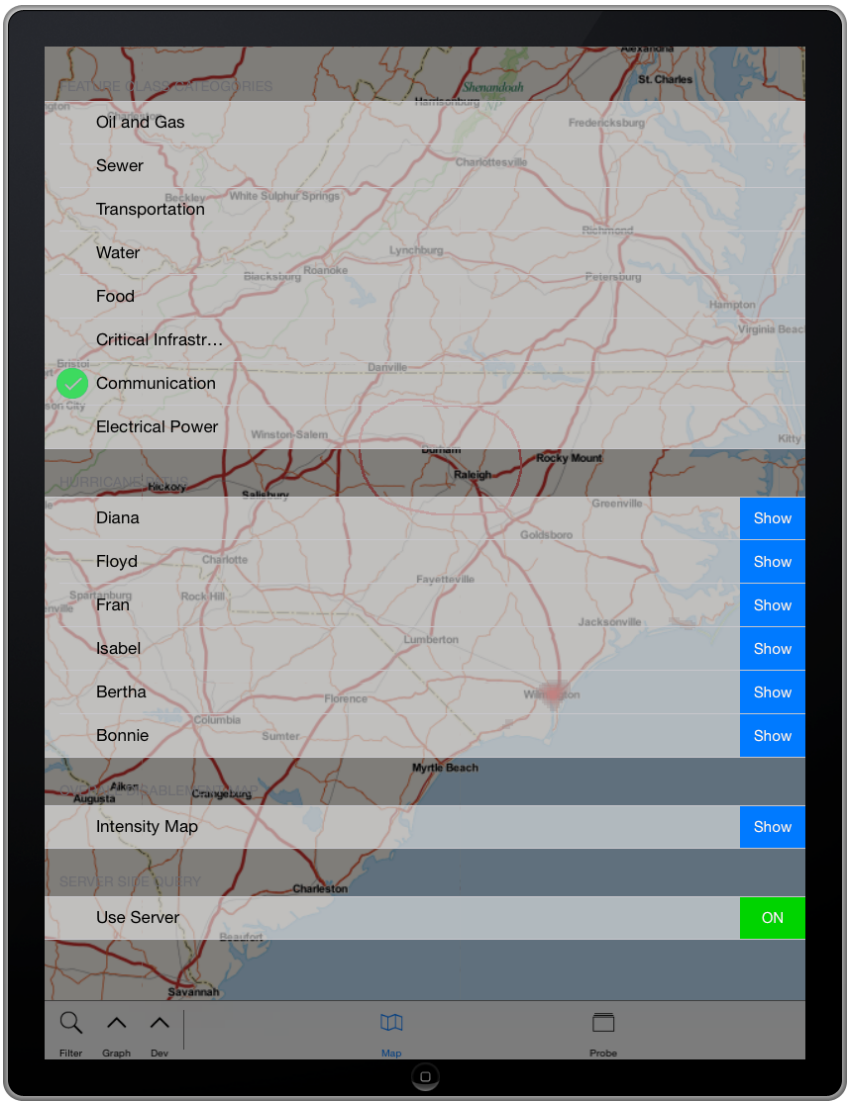


Figure 10. The Filter Overlay in our system. User can use this view to focus on different feature groups (e.g., Communication) or individual hurricane path

is necessary in order to maximize every inch of the screen. This allows the user to intuitively make selections and understand the effects of their changes.

Given our focus on the disaster response domain, we choose the geospatial view as the main entry view where all ensembles are presented at the initial stage. The coordinated views for temporal analysis and filtering are split into two transparent overlays. Specifically, the user is able to select either a filter view or a temporal view that contains a transparent overlay in order to see any changes made to the spatial representation. The user can also very quickly hide the overlays with a simple swipe gesture in order to return to spatial navigation.

5. CASE STUDY AND EXPERT FEEDBACK

Evaluating our system and collecting results through a traditional user study is challenging because the focus of our work is centered on both domain experts and responders. Such evaluation requires a group of responders from multiple domains working on a large collaborative disaster response exercise. Due to the limited availability of responders, we have demonstrated our application through simulated scenarios to determine our systems usability, effectiveness, and need for improvements. In this process, we conducted two evaluation sessions with three city

emergency responders and four power grid managers from a large regional power company. We thus are getting feedback from both responders and domain experts.

5.1 Forecasting Events prior to Hurricane Season

Scenario: The center focus of this scenario was Wilmington, NC, a coastal city that is historically vulnerable to hurricanes. Specifically, over 1 million cascading infrastructure events were examined over 8 hurricane paths that had directly passed through the greater NC region. As shown in Figure 9, our mobile ensemble visualization provided users an exploratory environment to identify where and what are the most vulnerable facilities.

In this case, an emergency responder was interested in isolating critical infrastructure that would likely cause finical distress if it were destroyed or rendered inoperable from a hurricane. The city planners need to specifically find these critical points like power substations and water services that could be very costly to the community if they are knocked out. They quickly examined the region using the probing function by drawing a bounding area around the different areas of the city. This action brought up a secondary window with the time distribution of events in that region, as shown in Figure 9. With multiple probing windows opened within the different quadrants of the city, the planners were able to see the aggregate view of the outages over time, but more importantly he was able to compare the peak times of outages side-by-side. Within the same window the planners were able to view the complete break down of events, which contains outages of airports, ports, highway bridges, and train stations. To one planners surprise, the airport in this region shows a significant outage at the early stages of our simulations, suggesting a high chance for this facility to be disabled given the simulated hurricane paths. This is just the sort of unexpected but quite important discovery that our system supports.

The planner hypothesized this may be caused by road blockage and direct impact and navigated through the interface and selected the detailed breakdown for the airport. He was surprised to see the report on what critical events caused the outage, as shown in Figure 7 (B); although it complies with his experience that the outage may be caused by direct impact (44%), there still was an unexpected 51.2% outage caused by Electric Service failure. This provided him more insights and helped him to take action on investigating weak links in power connections to the airport by working with power company participants.

User Feedback: One of our architecture's advantages is the ability to compute massive amount of physically-based infrastructure simulations ahead of a hurricane season. All collaborators consider this is of great value as they can start their planning process a couple of months ahead and become fairly comfortable with the range of resiliency for their infrastructures. As summarized by one of the emergency planners, "this system gives us the ability to look forward and be prepared. It could help us to find a vulnerable building and plan ahead the scenarios when a hurricane actually hits".

However, since our analysis is built upon simulations, managers from the energy company suggested further extensions to fuse heterogeneous datasets into the mobile ensemble visualization. In particular, they are interested in learning how we could effectively associate public information, such as census and demographics data.

5.2 Investigating Cascading Effort of Infrastructure Breakdowns

Scenario: Our mobile visual analytics environment enables users to explore and follow the impact caused by the evolution of a natural disaster. A city planner using the system inspected a couple of possible hurricane paths that made a pass through the Wilmington area (Figure 5.

Using the direct filtering methods, the user can quickly narrow down the analysis scope to the paths within close proximity to the city and examine critical areas along it. As a transportation expert, her area of focus is on how to bring back the transportation infrastructure after a catastrophic impact. Hence she further filtered down with transportation infrastructure and noticed an interesting spreading pattern that presents an elongated trailing effect as the effect of the hurricane is still impacting the infrastructure even a month after the initial landfall. She was surprised to see this trailing effect and suspected this may due to a less robust road structure. Indeed, a quick examination of the transportation network near the coast revealed that this may be one of the possible cascading effects due to the less redundant pathways from the outer coastal region to the inland. Once the hurricane passes through, getting relief and assistance back into the city requires outlets for transportation.

Our system allows the user to then apply a specific filter just on transportation to see just those particular events in conjunction with the hurricane paths selected already. In this scenario, the mobile interface permits the user to focus on a specific infrastructure breakdown of her particular analysis interests.

User Feedback: One of the benefits that all these users, both electric system experts and emergency planners, see in our architecture is its capability in helping to depict the overall impact on critical infrastructure as well as directly assess individual commodities (e.g., transportation structure). Using this visual interface, they can select specific hurricane paths that would pass through a certain city. Especially to planners, who are responsible for correlating information from various simulation models, the capability to identify and filter the infrastructures from a massive simulations space is of great value. As commented by a power grid manager that "...this is very useful for me to follow and compare my power lines with possible hurricane paths. I can then re-route my power transmissions away from the danger zone."

Several collaborators mentioned the need to conduct search-by-example functions due to the limited time constraint as a disaster is unfolding. As stated by an emergency response manager that, "it would be great if we can select one [impact] pattern, and the system can automatically suggest other similar ones...this would save a lot of time for me to navigate through the whole dataset." This requires our architecture to be able to perform comprehensive ensemble structuring, producing a more similar space for all the ensemble structure. His comment is well received, and we are working extensively on researching a quantifiable ensemble structure for ascertaining where attention is needed and resources should be deployed.

It is important to note that these rather complex analyses, building of hypotheses, and checking of the hypotheses were done on the mobile device in both cases. This indicates that the ensemble visualizer can be carried around for impromptu meetings among responders or planners and can also be carried into the field. Being able to look in detail at vulnerability of infrastructure in the field is quite important for both planning and response. Detailed actions can then be taken on-the-spot. In an actual emergency, a response team would be on hand to carry them out.

There is a final important outcome from our studies. We have shown that the mobile interface can be used to derive actionable knowledge and make decisions by both experts and by non-expert users. In the case studies above the experts were electric system managers, who would know a lot about the basis for the underlying critical infrastructure simulations and the non-experts were emergency planners, who know little about these simulations but must still make key decisions based on them. Both types of users were able to use the mobile interface effectively with a small amount of initial training. The details of the complex system outlined in Section 4 were hidden from both types of users. Yet they were available to the experts as needed. Indeed the experts could even control some details of the simulation through an interface that makes this complex activity much more straightforward. This important result shows the broad utility of the system we have set up. It can be used effectively by the range of people who will needs its capabilities in planning for or responding to large scale disasters, and it can support partnership and coordination between them.

6. LIMITATIONS AND FUTURE WORK

There are limitations to this research that must be addressed. Although our research was conducted towards the goal of ensemble analysis of simulation results, the complexity, the scalability and the mobility needs of the targeted dataset exacerbated the challenges on a computational-level and visualization-level. Especially, we found it very challenging when designing the multiple-coordinated views on a mobile device. While such research activity is upcoming, there is no gold standard to follow when considering the small screen space in mobile devices. We did attempt to mitigate the challenge a) by following mobile emergency design practices that we have accumulated experience by[7] and b) by projecting ensemble abstractions to spatial-temporal visualizations that can best utilize the mobile displays. Nevertheless, multiple coordinated visualization in different domains still requires additional design considerations. We acknowledge this challenge and are working closely with domain users to provide a more customized visual interface.

In addition, our scenarios and expert feedback are limited to the available resources within the project. A more in-depth assessment of our system utility can only be derived when it's deployed in the field with the

responders. Further, in our study the effects of time pressure were not considered, which may have significant influence on decision makers (Section 5.2). We will look into incorporating techniques, such as search-by-example, to reduce user the time pressures.

In future work we would like to run simulations in the cloud that are driven from a mobile device. There are a number of limitations that prevent us from generating new simulations in real time. The first limitation is a computational problem. Running these exhaustive simulations is very time-consuming and would require new more efficient methods to run in real time. Secondly, we would have to develop effective methods for generating a simulation model on a mobile device and then communicate that model to our simulation environment in order to evaluate the simulation. Currently the simulation environment is a closed system that is self contained. Additional modifications and designs would have to be made to allow for two way communication with the simulation environment.

We recognize these limitations and consider the support for disaster forecast and preparation as an important visualization, analytics, and interaction research topic. We expect the presented approach illuminates the role that mobile ensemble interfaces play in such complex problem-solving environments and provides a platform for actual use.

7. CONCLUSION

Understanding large cascading simulation datasets is vital in disaster monitoring and emergency response. But, gaining insight from these simulations requires extensive tools and analysis in order to provide planners, emergency responders, and electric system experts with valuable information. To gain enough insights for actionable knowledge, we utilize ensemble visualization of a large scale simulation space. We have developed a general critical infrastructure simulation and analysis system for situationally aware emergency response during natural disasters. Our system demonstrates a scalable visual analytics infrastructure with mobile interface for analysis, visualization and interaction with large-scale simulation results in order to better understand their inherent structure and forecast capabilities. The utility and efficacy of our research has been evaluated by domain practitioners and disaster response managers.

ACKNOWLEDGMENTS

This material is based upon work supported by the U.S. Department of Homeland Security under Grant Award Number 2009-ST-061-CI0001-06. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security.

REFERENCES

[1] Vincent, L. and Soille, P., "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Trans. Pattern Anal. Mach. Intell.* **13**(6), 583–598 (1991).

[2] Pascucci, V., Laney, D. E., Frank, R., Scorzelli, G., Linsen, L., Hamann, B., and Gygi, F., "Real-time monitoring of large scientific simulations," in [*Proceedings of the 18-th annual ACM Symposium on Applied Computing*], 194–198 (March 2003).

[3] Edelsbrunner, H. and Mücke, E. P., "Simulation of simplicity: A technique to cope with degenerate cases in geometric algorithms," *ACM Transactions on Graphics* **9**, 66–104 (1990).

[4] Santos, E., Freire, J., Silva, C., Khan, A., Tierny, J., Grimm, B., Lins, L., Pascucci, V., Klasky", S. A., Barreto, R. D., and Podhorszki, N., "Enabling advanced visualization tools in a simulation monitoring system," in [*Proceedings of the 5th IEEE International Conference on e-Science*], 358–365, IEEE (December 2009).

[5] Dietrich, J., Trahan, C., Howard, M., Fleming, J., Weaver, R., Tanaka, S., Yu, L., Jr., R. L., C.N, Dawson, J., Westerink, Wells, G., Lu, A., Vega, K., Kubach, A., Dresback, K., Kolar, R., Kaiser, C., and Twilley, R., "Surface trajectories of oil transport along the northern coastline of the gulf of mexico," in [*Continental Shelf Research*], *Elsevier* (2012).

[6] Horne, G., "Beyond point estimates: Operational synthesis and data farming," *Maneuver Warfare Science* (2001).

[7] Guest, J., Eaglin, T., Subramanian, K., and Ribarsky, W., "Visual analysis of situationally aware building evacuations," *Proc. SPIE* **8654**, 86540G–86540G–14 (2013).

[8] Apple, Inc., "ios human interface guidelines," (Sep 2013).

[9] Safford, T., Thompson, J., and Scholz, P., "Storm surge tools and information: A user needs assessment," NOAA Coastal Services Center.

[10] Potter, K., Wilson, A., Bremer, P.-T., Williams, D., Pascucci, V., and Johnson, C., "A flexible approach for the statistical visualization of ensemble data," in [*Proceedings of IEEE Workshop on Knowledge Discovery from Climate Data: Prediction, Extremes, and Impacts*], (December 2009).

[11] Mascarenhas, A., Grout, R. W., Bremer, P.-T., Hawkes, E. R., Pascucci, V., and Chen, J., [*Topological feature extraction for comparison of terascale combustion simulation data*], Mathematics and Visualization, Springer (2010). to appear.

[12] Bürger, R. and Hauser, H., "Visualization of multi-variate scientific data," in [*Eurographics 2007 STAR*], 117–134 (2007).

[13] Buja, A., Cook, D., and Swayne, D. F., "Interactive high-dimensional data visualization," *Journal of Computational and Graphical Statistics* **5**, 78–99 (1996).

[14] Feild, H. and Emery, K., "An uncertainty analysis of the spectral correction factor," in [*Photovoltaic Specialists Conference, 1993., Conference Record of the Twenty Third IEEE*], 1180–1187 (May 10–14 1993).

[15] Robert, C. P. and Casella, G., [*Monte Carlo statistical methods*], Springer (2004).

[16] Mileyko, Y., Mukherjee, S., and Harer, J., "Probability measures on the space of persistence diagrams," *Inverse Problems* **27**, 124007 (2012).

[17] Turner, K., Mileyko, Y., Mukherjee, S., and Harer, J., "Fréchet means for distributions of persistence diagrams," *Manuscript* (2012).

[18] Nocke, T., Fleshig, M., and Böhm, U., "Visual exploration and evaluation of climate-related simulation data," in [*IEEE 2007 Water Simulation Conference*], 703–711 (2007).

[19] Hubbard, B., Kellum, J., Pual, B., Santek, D., and Battaiola, A., "Vis5d." http://vis5d.sourceforge.net.

[20] Sanyal, J., Zhang, S., Dyer, J., Mercer, A., Amburn, P., and Moorhead, R., "Noodles: A tool for visualization of numerical weather model ensemble uncertainty," *IEEE Transactions on Visualization and Computer Graphics* **16**, 1421–1430 (2010).

[21] Potter, K., Wilson, A., Bremer, P.-T., Williams, D., Doutriaux, C., Pascucci, V., and Johnson, C. R., "Ensemble-vis: A framework for the statistical visualization of ensemble data," *International Conference on Data Mining* **0**, 233–240 (2009).

[22] Wong, P. C. and Bergeron, R. D., "30 years of multidimensional multivariate visualization," in [*Scientific Visualization, Overviews, Methodologies, and Techniques*], 3–33, IEEE Computer Society, Washington, DC, USA (1997).

[23] Samtaney, R., Silver, D., Zabusky, N., and Cao, J., "Visualizing features and tracking their evolution," *Computer* **27**, 20–27 (July 1994).

[24] Butkiewicz, T., Chang, R., Wartell, Z., and Ribarsky, W., "Visual analysis and semantic exploration of urban lidar change detection," *Comput. Graph. Forum* **27**, 903–910 (2008).

[25] Ni, D., Chui, Y., Qu, Y., Yang, X., Qin, J., Wong, T., Ho, S., and Heng, P., "Reconstruction of volumetric ultrasound panorama based on improved 3D SIFT," *Computerized Medical Imaging and Graphics* **33**, 559–566 (2009).

[26] Eaglin, T., Subramanian, K., and Payton, J., "3d modeling by the masses: A mobile app for modeling buildings," in [*Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on*], 315–317 (2013).

[27] Anselin, L., Syabri, I., and Smirov, O., "Visualizing multivariate spatial correlation with dynamically linked windows," in [*New Tools for Spatial Data Analysis: Proceedings of the Specialist Meeting*], (2002).

[28] Federal Emergency Management Agency, "http://www.fema.gov/hazus."

[29] The Federal Communications Commission, "http://www.fcc.gov/data/download-fcc-datasets."

[30] Environmental Systems Research Institute, "Arcgis desktop," (March 2014).

[31] Google Inc. http://map.google.com.

[32] Mittelstdt, S., Spretke, D., Thom, D., Jckle, D., Karsten, A., and Keim, D. A., "Situational Awareness for Critical Infrastructures and Decision Support," in [*Proceedings NATO STO IST-116 Symposium on Visual Analytics*], (2013).

[33] Maciejewski, R., Rudolph, S., Hafen, R., Abusalah, A., Yakout, M., Ouzzani, M., Cleveland, W. S., Grannis, S. J., and Ebert, D. S., "A visual analytics approach to understanding spatiotemporal hotspots," *IEEE Transactions on Visualization and Computer Graphics* **16**, 205–220 (Mar. 2010).

[34] Wang, X., Dou, W., Butkiewicz, T., Bier, E., and Ribarsky, W., "A two-stage framework for designing visual analytics system in organizational environments," in [*Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on*], 251 –260 (oct. 2011).

[35] Sedlmair, M., Meyer, M., and Munzner, T., "Design Study Methodology: Reflections from the Trenches and the Stacks," *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)* **18**(12), 2431–2440 (2012).

[36] Tile Mill, "Tile mill https://www.mapbox.com/tilemill/."

[37] Robertson, G., Cameron, K., Czerwinski, M., and Robbins, D., "Animated visualization of multiple intersecting hierarchies," *Information Visualization* **1**, 50–65 (Mar. 2002).

[38] Roberts, J. C., "State of the art: Coordinated and multiple views in exploratory visualization," in [*Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization*], 61–71 (2007).

Interactive Visual Analytics in Support of Image-Encoded LIDAR Analysis

Todd Eaglin, Xiaoyu Wang, and Bill Ribarsky

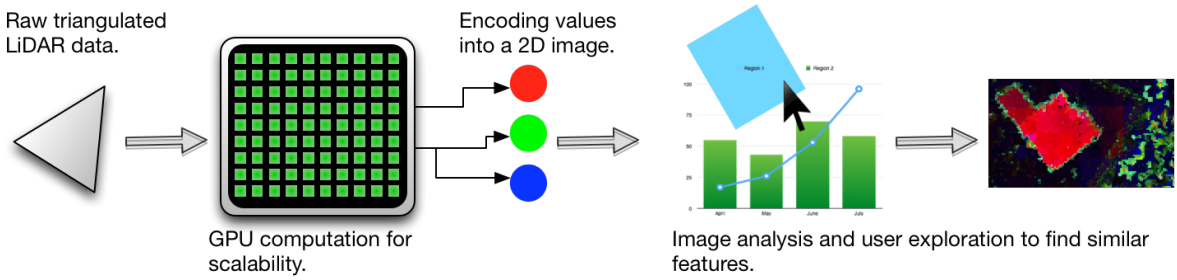


Fig. 1. The overall process of our system and methods for taking raw triangulated LIDAR data and transforming it into a form that is scalable, interactive, and easily searchable with by users for interactive visual analytics.

Abstract—LiDAR data is a significant resource for identifying similar geospatial features in urban planning, land use analysis, emergency response, and other applications. Traditionally LIDAR is analyzed through manual process, which is a very challenging task due to the need to identify similarities over a growing size and complexity of data. To alleviate this challenge, we designed and developed a GPU-powered visual image analytics system to handle this operation at large scale. Our system encodes human-freeform-LiDAR selection into 2D images through an autonomous image analysis process that matches selected areas of interest. To ensure the systems practicality in handling hundreds of stitched LiDAR patches, we have scaled up our algorithms through a series of parallelized GPU processing, analyzing, and encoding methods. We conducted informal user studies to assess the utility and usability of the system.

Index Terms—Visualization, LIDAR, GPU

1 Introduction

LIDAR, a remote sensing technology, has become a crucial instrument for us to extract and monitor up-to-date, accurate geographic information about areas of interests. Professionals involved in a variety of industries and applications (like disaster management) are increasingly utilizing three-dimensional sources of information to create photorealistic 3D visualizations, extract 3D features and export products to geospatial tools to help understand geospatial features in both urban planning [5] and emergency responding [7].

Especially in emergency management and disaster response, LIDAR presents a more effective data source to locate weak points. For instance, Kwan et al. [12] used about 50 million LIDAR points to identify blockages in the transportation network caused by Hurricane Katrina. Whereas, Clasen et al [6] further examined the potential areas where LIDAR data can be used for disaster and emergency planning, including understanding how slope of the terrain affects landslides, analyzing tree maps for fire disasters, and more importantly finding suitable areas for rescuer deployment and helicopter landings.

Zerger et al [18] examined the current technical limitations with GIS in emergency situations. They discovered that existing systems were unable to provide details in real-time due to limited processing power and the size of data. Database queries were extremely time consuming and unsuitable for real-time planning.

While the utility of LIDAR can be significant, analyzing it effectively over a larger area still imposes a significant challenge. Based on our collaboration with North Carolina Department of Transportation, we observed the following challenges:

- *Scale* - Due to the nature of LIDAR it can grow exponentially when examining larger spatial regions, at higher resolutions, and at different points in time. The files can be hard to manage; even loading and viewing them can be a problem. Collecting different sources of data, processing them with a myriad of different tools, and converting LIDAR coordinate systems can also be quite time-consuming. Kwan et al. [12] used a data set of about 50 million LIDAR points, but they do not mention the scalability of their work and how quickly it takes to process.
- *Interactions* - There is a need for being able to analyze and manage LIDAR data in realtime. Otherwise, it's difficult to do comparative feature analysis or disseminate the results to responders. There is a need for interactive tools for analysis and searching to solve these problems.
- *The lack of searchable content*. To be able to fully and effectively analyze the LIDAR data, the system must support: finding areas of similar elevation; finding buildings of similar size or type; finding road networks and their structures (e.g., bridges, embankments, ditches, etc.). In addition one should be able to find special relationships such as roads near areas of high vegetation; roads near coastal areas.. . .

To alleviate these challenges and make LIDAR analysis more interactive and effective, we present a GPU-powered visual image analytics system that processes, analyzes, searches LIDAR data at scale. Our system encodes human-freeform selection into 2D images through an autonomous image analysis process that matches selected areas of interest. By leveraging the ability of users to create meaning and relationships in the data, our approach becomes the starting point for

- *Todd Eaglin. E-mail: teaglin@uncc.edu.*
- *Xiaoyu Wang E-mail: Xiaoyu.Wang@uncc.edu.*
- *Bill Ribarsky E-mail: ribarsky@uncc.edu.*

assisting them in understanding these datasets for a multitude of problems. Our work focuses on how a user can be helped to identify features and find similar ones via intelligent searching amongst millions and millions of LIDAR data points.

The rest of the paper is structured as follow: in section 2 we discuss in depth why we choose to use image analysis and the benefits of using that approach. We cover the computational and space complexity of an image analysis approach. We also discuss how it ties in with visual analytics and the overall goal of incorporating a user's knowledge to drive the analysis. In Section 3 we present our method for encoding details from LIDAR data into a 2D image. We discuss the algorithms that we used in the context of GPU computing as a means to accelerate user interactivity. Our Visual Analytics system is presented in Section 4; we present use cases an informal user study in Section 5. We wrap up with conclusions in Section **??**.

2 Related Work

In this section we discuss our motivation for tackling this problem and why we have chosen image analysis and how that plays into our larger goal of allowing real time interaction and analysis of LIDAR data. Since our approach is multidisciplinary, we cover the recent work in both image analysis with relation to LIDAR visualizations. For example, Richter et al [13] examined a novel use of improving LIDAR visualization using GPU acceleration and out of core rendering. Butkiewicz et al [4] explored the combination of visual analytics and LIDAR to detect temporal changes in a city environment. The authors looked at finding the differences between different time periods of LIDAR data using the 3D geometry created from a LIDAR point cloud. Using the computed geometry provided more accuracy than a point cloud using a grid approach with nearest neighbor and outputting that as a 2D image. In our approach we create 3D geometry to provide more accuracy and details, but we return to using 2D images by encoding the geometry back into an image. We do this for several reasons that we discuss in Section 3.

Butkiewicz et al [4] did not mention anything about scalability since they compared unique spatial regions in time and did not compare any deltas between regions. This resulted in a much smaller number of overall comparisons. Our approach allows a user to analyze two different regions regardless of time. The authors concluded that there is still much work to be done in the area of algorithms for handling issues of matching or issues with users not being able to define filter metrics. Blaschke et al [2] surveyed the existing research in image analysis for remote sensing; they outlined some of the benefits to using image segmentation.

A lot of work has been done on classification in LIDAR data and extraction of specific features like buildings, as described by Hermosilla et al [10], who utilized image thresholding using two specific values. The first threshold is value based on the minimum height of a particular building and the second value indicates the presence of vegetation. Their results were quite strong, using image thresholding, but this was only when the thresholding values were adjusted to the specific type of environment in the LIDAR data. We expand on this work in several ways. Firstly, we add more thresholding values by converting a LIDAR point cloud into a RGB image from a triangulated 3D mesh. During this process of creating a new image from the LIDAR point cloud, we encode high quality information about unique traits, but more importantly we encode them in a way that they are properly adjusted so thresholding can be done more efficiently and with a clearer understanding of the result. We discuss this in detail in Section 3.

2.1 Why Image Analysis?

Our data set consists of about fifty Digital Surface Model (DSM, ground elevation plus all features, such as buildings, on it) LIDAR patches along the coast around Wilmington, NC. Each patch is made up of X, Y, Z locations that denote the geographical position and the elevation of the terrain at that position. Each patch in the data set contains between 1 million and 4 million points. In total we have approximately 100 million points.

Computation Evaluation

Apprx 1 million points per patch	1 Patch	5 Patchs	10 Patchs	25 Patchs	50 Patchs
CPU - Single Thread	1m 28s	7m 20s	14m 50s	32m 30s	1h 6m
CPU - Multi Thread (12 threads)	1m 28s	2m 3s	3m 39s	8m 53s	19m 45s
CPU - Multi Thread (12 threads) + GPU	18s	43s	1m 20s	3m 16s	6m 34s

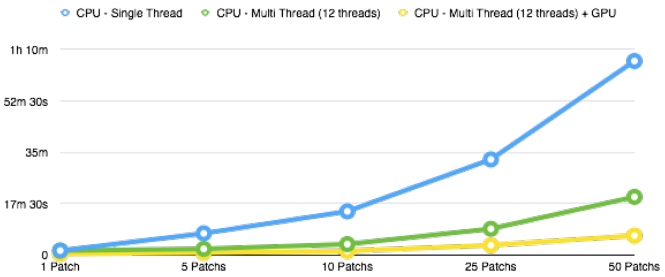


Fig. 2. Timing results using three different metrics. The first is a single thread CPU implementation. The second is a multi-thread CPU implementation. Lastly, is our implementation of multi-thread CPU and GPU. The benchmark was performed on a 3.5 GHz 6-Core Intel Xeon E5, with a GTX 960 Nividia GPU.

To ensure the real time interaction and analysis of LIDAR data, we turn to research in image analysis. Primarily, we utilize image analysis to handle both the space and time complexity of LIDAR data and automation.

We are also using image analysis because the area of digital image processing/computer vision is well-established with good techniques for handling images and finding unique features. Since we compute the resulting image this allows us complete control over the output making image analysis an even better choice.

2.1.1 Space and Time Complexity

LIDAR data is complex and very large. So doing realtime interaction can be difficult. But, converting the data to an image and rendering it makes it easier to handle. Using images is also a great means of storing the resulting data after processing. In most cases the resulting image is dramatically smaller in memory footprint than the original LIDAR data used to produce it. We are essentially compressing the original data while adding even more information.

Using our data set we have multiple raw patches with four million points. The memory footprint of one of those patches is about 200MBs. Converting it to an image not only adds more information but dramatically decreases the size by almost a factor of one hundred to 2MBs. Since the image size is fixed that means it will scale very well.

We ran three benchmarks using different core combinations to evaluate the scalability and performance of the algorithm for use in interactive applications. To tightly control the benchmark tests we used the same LIDAR patch, containing approximately one million points and had the application load it multiple times based on the number of patches being tested.

2.1.2 Automation

Automation is an important tool in analyzing large data sets because it helps a user focus on what matters. It delegates the complexities and challenges of the data to the computer, while letting a user focus on the decision making process.

Semi-automated techniques while quite useful still require lots of existing data and training sets to get meaningful results. This can be time consuming not just in teaching a system, but collecting the data in the first place.

We achieve automation by intelligently selecting specific values so that we encode enough information at each pixel. The better the information we can encode into the resulting image the more accurate

results we can achieve. We encode high quality sources of information, then we can use image analysis techniques to quickly compare different sections without requiring user training.

3 Algorithm and Methods

In this section we will detail the overall design and methods we used for our system. The first part covers how we encode unique details into an image, what those unique values are, and how we calculate those unique values. The second part covers the actual system implementation and the technologies we use to build the system.

3.1 Process and Encoding Values

In this section we detail the process of taking a LIDAR point cloud and transforming it into a custom image with specific r,g,b values in which we have encoded unique details.

The first step of the process is taking the point cloud and generating a triangulate continuous mesh using Delaunay triangulation. We then compute specific values using these triangles. We used prior work done in visually encoding unique attributes from [15], Composite Density Maps for Multivariate Trajectories. This paper developed a flexible architecture for visually analyzing attributes that might reveal patterns. The authors were able to easily modify and calculate attributes and composite them into images in order to find unique patterns. In our work we propose three attributes that we believe were the most helpful in our analysis use cases, but the system is flexible enough that additional analysis techniques could be introduced or added to provide a new dimension of understanding or pattern analysis.

As part of our implementation we process and load each patch independently. To do this efficiently, we must manage the data in parallel fashion using OpenCL/OpenGL and other high performance technologies. We've placed all of our visual encoding operations onto the GPU using a GPU quad tree with a mixed approach using spatial hashing for our specific problem. As we discuss later we do this as part of the analysis step to save memory and computation. We begin by constructing the quad tree and spatial hash of all the triangles within the mesh. After we have completed this operation we pass these parameters into the GPU: triangle indices, vertex positions, vertex color, quad tree, hashed cells, and visualization parameters. Due to the synergy between GPU computation and rendering technologies we send all this information together because we can then instantly render the triangle data with the computed color data since it already exists on the GPU. This saves a significant amount of time due to the slow nature of copying data to and from system memory to GPU memory.

3.1.1 Quad Tree/Spatial Hashing Construction

Due to the sheer scale of our data utilizing GPU computation is a necessity. We use a mixed approach combining spatial hashing with a quad tree. We start off as usual recursively subdividing triangles in 4 quadrants storing each quad in a heap array for direct ingestion by the GPU. Once we subdivide to a point where a quadrant contains 20 or less triangles we stop. This is where it gets interesting. Since GPU languages are C one has to be very careful with how one represents data since objects can't be passed. The memory is also limited, so large amounts of data can't be stored. [9]

At the 20 triangles or lower point a spatial hash cell is created that contains an index to each of those triangles from the global list of triangles. Then this cell is added to a global array and the index of its location is stored in that leaf quadrant. This produces an in order heap of just the triangles collected in the cells based on the leaves in the quad tree. As the recursion cycles back from the leaves to the root, a cell index range for each quadrant is built. This is possible because the leavess are in order. So at any given quadrant in the tree all the triangles can be accessed in roughly in roughly O(n) time because everything is indexed into an array. We use 20 triangles as the cut off point for a spatial hash cell because there is some memory overhead to maintain each cell. Since GPUs are limited in memory we chose this cut off point specifically for our setup. Lowering the cut off point would decrease the computation time since each triangulated patch is on the order of a few million triangles, but it would increase the memory overhead.

3.1.2 Local Normalized Elevation

We compute the normalized height of each vertex of each triangle based on it's surrounding triangles. This allows for very easy comparison of a specific region of the resulting image with any other area in the image. Computing the normalized height is the hardest part of all the values we encode. Since we are normalizing the elevation to values between [0.0,1.0] it is very sensitive and difficult to quantify if there is a gradual slope in the terrain. For example suppose there are two buildings with the same height at two opposite ends on a LIDAR tile and there is a gradual slope in the terrain between them. Normalizing the elevation will result in one building being drastically higher then the other when both buildings are the same height.

Our work focuses on using just DSM LIDAR. We do not directly calculate the DTM (that is, the ground terrain surface without buildings or other features) from the LIDAR data to create the nDSM (normalized digital surface model, or the elevation of all objects on a flat surface) It is calculated by finding the difference between DTM and DSM. Doing so would require another computational step and require more memory. Instead we do normalization as part of the analysis process. We looked at several approaches. The first was the normalizing approach described in the previous paragraph, but this produced undesirable results. As mentioned buildings of the same height were being miscalculated due to variances in overall elevation. The second approach was a brute force comparison using a defined radius to find nearby triangles, but due the massive size of the data this operation was extremely slow and far from ideal. The third approach was sampling a large set of random triangles. This approach actually produced decent results and was quite fast, but it did not always work.

We then explored using a quad tree and using the minimum and maximum of the leaf nodes in the tree. This worked, it was fast, but it did not handle cases of very flat areas. For example our data covers a coastal region with a wide mix of elevation types from water to downtown cities. The normalized elevation for the water areas was heavily miscalculated. To fix this issue we include one more step. In order to account for variances in gradual terrain slope we use a statistical approach to measure the variance compared to the minimum and maximum elevation of the entire LIDAR patch. We do this by first selecting a specific triangle and iterating to the lowest quadrant containing it in our quad tree. If the variance in elevation does not meet a specific deviation from the elevation of the entire LIDAR patch then we step up in the quad tree to the parent and do the calculation again until we eventually get to a quadrant that contains enough variance. This iterative approach is similar to the method described in [8], but instead of iteratively using smaller and smaller windows we use the quad tree to find the best defined area for normalization.

The normalized height is the single most crucial value that we encode for doing any analysis. It also drives the rest of the values we encode. Since we select a region with enough statistical variance in elevation we use that region to calculate any comparison analysis that we encode.

3.1.3 Surface Normal

The surface normal is the vector of the plane created by the triangle. We have to encode this three dimensional vector to a one dimensional value. We do this by taking a uniform directional vector and computing the dot product of it and the surface normal. This creates a one dimensional value between 0.0 and 1.0 that gives us the slope of the surface. Computing the slope of the surface allows us to very easily distinguish steep surfaces like buildings and trees.

3.1.4 Navigation Mesh and Drive-ability

During our initial work for determining values, we found it difficult to detect a road network using all the previous methods. Our initial analysis was for LIDAR patches in coastal regions. We found thatdue to the nature of the elevation of the ground, very minimal building

placement, and vegetationit can be difficult to reliably detect roads from some types of LIDAR.

One option is to use existing road vector maps; while this would produce the best results it is not always feasible. The points in time when the LIDAR was collected and when the road vector maps were created can be different. Analyzing the differences between different periods in time, especially in rural and under-developed areas would require collecting older road vector maps and might not be possible in all cases. We wanted a consistent solution based on the LIDAR itself.

The solution we explored was generating a navigation mesh based on the geometry of the resulting triangulated LIDAR data [11]. We explored navigation mesh creation using an image-based approach with LIDAR in [1]. But, the work they presented was done in city landscapes and there is no mention of rural environments or any such results. In addition, most navigation mesh research aims to produce a simplified graph network for performing path planning. Our goal is not to produce an optimized graph, but to create an estimation metric for determining a road network. Therefore, we are not concerned about the complexity of the graph since we have no plans to perform path planning.

As part of the navigation mesh creation process we discard triangles with surface normals too steep for a road. As well we discard triangles that are too small for a car to pass on. During the creation process we discard any smaller disconnected sub graphs that are created. This results in a network that a car can physically drive on. While not completely accurate, it estimates a reliable road network that can be easily encoded into an image.

3.2 Image Rendering

After processing a LIDAR patch and calculating the encoded values, we store them as a color for each vertex of each triangle in the triangulated mesh. We then use this color data and render the triangulated mesh to an image using a fixed orthographic projection and scale across all the LIDAR patches to maintain a constant pixel to meter ratio. For the purpose of our work we constrain the image size to 1024x1024 pixels using 72 DPI and a bit depth of 24 RGB. This flexibility allows us to control many aspects of the image and the memory footprint. If we are trying to process and analyze large numbers of LIDAR patches to where storing all the resulting images would require too much space, we can adaptively reduce the size of the image. If we want to compare just a few patches and there is ample space, we can increase the image size and bit depth to give more accuracy.

4 Application

In this section we describe the visual analytics tool that we have developed using our techniques. In order to do searching and finding we use thresholding and image segmentation. A user first makes a selection for some type of feature (for example a building). We determine the most frequent color in that user's selection. Using this RGB value we then threshold all existing processed LIDAR images. Since we have encoded meaningful information into the processed images, using thresholding is a highly effective technique for finding areas of similar value based on the user's selection.

4.1 Interface

We present to the user two coordinated views. We followed the guide lines in for Multiple View Coordination in [14]. One view is a global fixed view that provides a complete overview of all the LIDAR tiles. The second is a zoom in detail view that allows a user to navigate around and zoom into specific areas. Together the views are connected. In the global view a bounding box represents the view bounds and position of the zoomed in view. Below the coordinated views is a scatter plot that displays all the similar feature results calculated from a selection.

4.2 Interaction and Analytics

We provide the user with two selection tools. The first is a bounding box selection and the second is a lasso tool that is similar to what is in

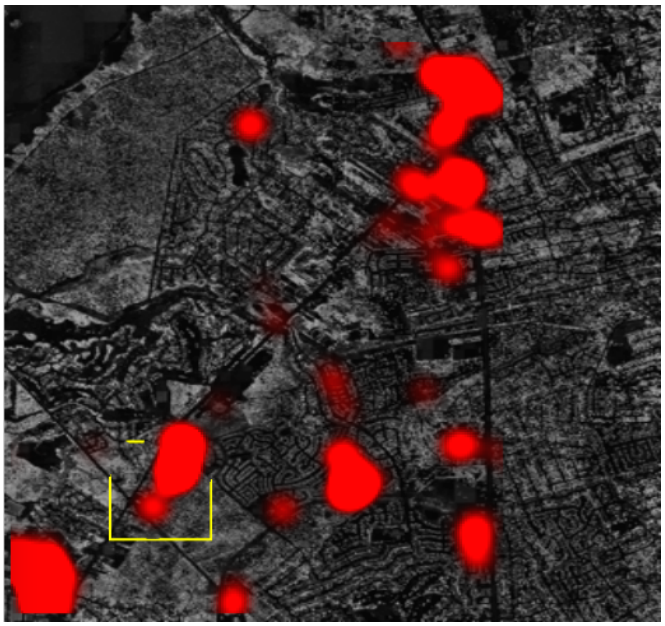


Fig. 3. One of the coordinated views. This view is the global view that shows a heatmap weighted by the similarity of the results. It also shows the bounding box for the the zoomed in detail view.

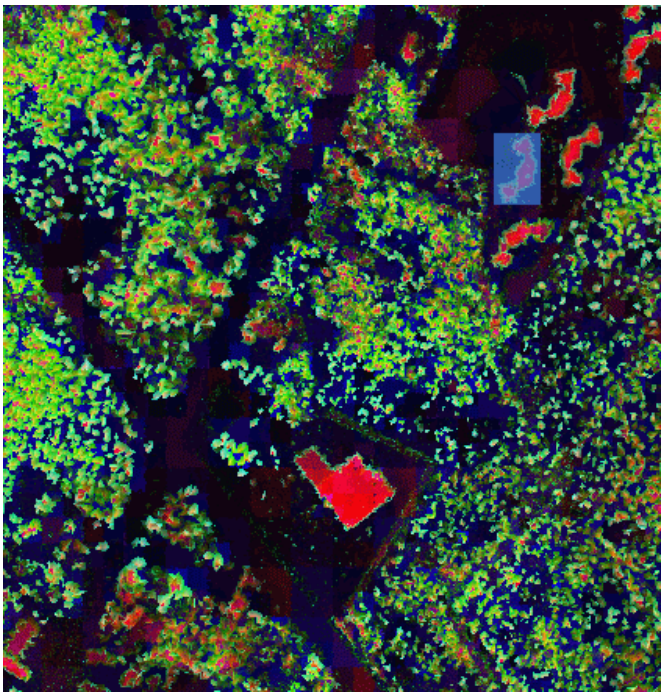


Fig. 4. One of the coordinated views. This view is the zoomed in detail view that allows users to make selections.

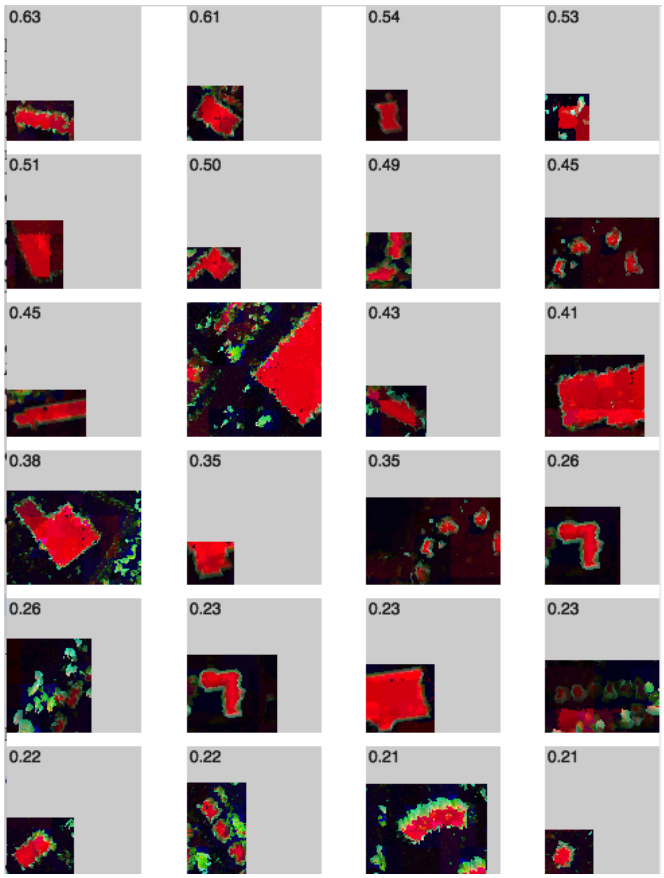


Fig. 5. The detailed list view, which shows the sorted results of a search. Each cell contains a ranking and an image of the feature that was found.

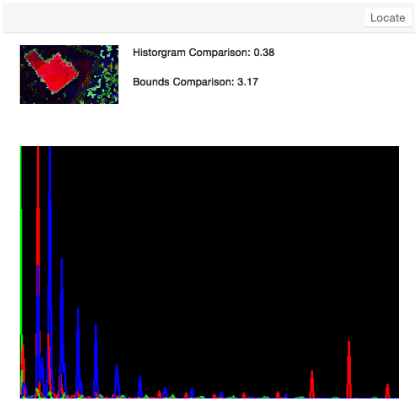


Fig. 6. The detailed feature view, which shows the historgram comparison and the bounds comparison. Below it shows a detailed break down of the histogram for that feature. At the top is a locate button that allows a user to locate the feature directly in the LIDAR patches.

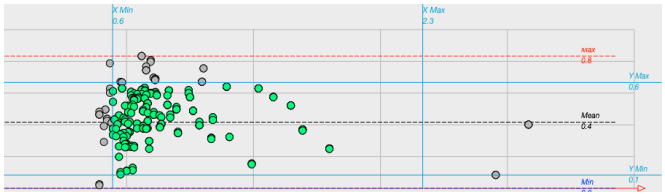


Fig. 7. An example scatter plot showing a selected subset of data from a search. The subset of data selected is represented by the highlight green points. The points in grey are unselected and filtered out.

$$d(H_1, H_2) = \frac{\sum_I (H_1(I) - \bar{H_1})(H_2(I) - \bar{H_2})}{\sqrt{\sum_I (H_1(I) - \bar{H_1})^2 \sum_I (H_2(I) - \bar{H_2})^2}}$$

Fig. 8. Histogram comparison function.

most image editing software applications. Once a user makes a selection we isolate that part of the image and run modified k-means clustering on the selection to extract the four most dominate colors. Due to its nature k-means does not produce the same results each time. This is significant because thresholding is very sensitive to small changes; different clusters from the k-means could result in quite different results. We explored solutions presented in [17]. Ultimately to fix this problem we ended up seeding each cluster using the same values each time based on the RGB channels. This produces the same results each time for the same selection.

Using each color we threshold the image using the GPU to speed up the process. Then we run boundary analysis and blob extraction. Each blob is then further analyzed and compared to the original selection through a histogram comparison and a bounds comparison. We plot each feature that is found in a scatter plot.

Along the x axis we plot how similar in width and height a specific feature is. A value of 1.0 would mean the feature is approximately the same size. A value of 2.0 would mean the feature is approximately twice as large as the user's selection and a value of 0.5 would mean the feature is about half the size of original selection. Along the y axis we plot the histogram comparison. We compute this value by creating a color histogram for the original selection and all of the results found. We then use the function in Figure 8 derived from OpenCV [3] to compute the correlation between two histograms. This metric ranges from 0.0 to 1.0 based on correlation. where 1.0 means the feature and selection have strongly similar histograms.

Within the scatter plot we display the max, mean, and minimum values from the resulting analysis. We also allow users to filter results further by sliding adjustable clipping lines that narrow results.

Any selection made in the scatter plot is then reflected in the detailed list view in Figure 5 and the global view in Figure 3. We used prior techniques to visualize spatial uncertainty that we have found effective in [7]. We incorporated glyphs to represent how similar a feature is to a user's selection. The glyph is represented as a circular ring that is added on top of the global view. The more complete the ring is the higher the similarity exists between that feature and the selection. A user can then further click on this glyph in the global view to reveal a detailed view of the feature.

Secondly we also incorporated a heatmap to quickly visualize hotspots. We use a kernel density estimation that is weighted using the histogram similarity. In this way users can quickly focus on regions with the highest similarity of results.

Lastly, we provide a concentrated list of the narrowed results from the scatter plot ranked by similarity. From this list users can select a specific feature that was found and visually examine it in Figure 6. They can locate it directly from the global view to find its origin. Also a user can create a new search using that feature to further refine what

they are looking for.

5 Case Study and Results

In this section we discuss the case study we ran and details of the three different tasks performed by participants. We also cover the results of our work, different methods we have tried, and any shortcomings we discovered.

As part of our case study we followed previous methods described in [16]. We selected 10 participants for an informal evaluation of the tool having them complete 3 basic tasks that focus on the visual analytics portion of our application. The participants in our evaluation were equally balanced between male and female. Their ages ranged from college age to senior citizen age. The professional background of our participants also covered a wide range from engineering to medical professional.

We set up three tasks ordered as to how a prospective user would use the application. In the first task we focused on discovery, exploration and selection. In the second task we took the prior selection from task 1 and built upon it by asking the participant to analyze the results of their selection. This included evaluating the spatial analysis results and detailed results. In the last task we asked participants to use the interactive tools to further drive the analysis by narrowing their original selection and analyzing the results from those interactions.

At the end of the session we followed up with participants with a questionnaire of 12 questions, 3 for each task and 3 detailing the overall usage of the tool. Each task had 3 questions with numerical scales to rank a particular sub-task. The final 3 questions ranked the overall learning experience and ease of use. The questionnaire also included a descriptive question asking which feature or function was most helpful to accomplish a task.

5.1 Task 1

In the first task we covered navigation and zooming as well as the use of the box and lasso tool. Initially we asked participants to navigate and focus on a particular building. We located for them on the global view a particular area of interest and asked them to navigate and focus on that region. We wanted to measure how easy it was for a participant to pick up the application and navigate to an area of interest. Secondly, we had them use both the box and lasso tool to make a selection around the area of interest. These two operations are the core of initiating the analysis of our tool so being able to very quickly navigate to areas of interest and make selections is vital to the analysis. All of the participants highly ranked the ease of use for navigating and focusing on a particular region.

Table 1. Task one result averages, ranked from 1 to 7. 1 being very difficult and 7 being very easy.

How easy was it to use navigating and zooming	6.5
How easy was it to use the box selection tool.	6.625
How easy was it to use the lasso selection tool.	6.125

5.2 Task 2

In the second task we had participants analyze the results of their prior selections from task 1. We first started by asking them to read from the scatter plot of results and detail for us the overall metrics of the results. This included identifying the complete range from the smallest result returned to the largest, as well as the least similar result to the most similar. We also asked them to tell us what the mean value was for each metric. The goal of this task was to get participants to analyze the results of their selection by evaluating the results returned in the scatter plot and doing visual analysis. As part of the visual analysis we had participants use the heatmap to locate regions and then had them perform basic interaction with the annotations to find results with spatial context.

We asked participants to use the heatmap overlay to tell us where there are areas of similar features. We then had the participants navigate to those areas. Next, we asked participants to switch from the heatmap overlay to the annotation overlay. We then asked participants to select one of the annotations in the global view in the region they had just navigated to and examine the result.

As part of our evaluation we had participants rank the ease of use in using the scatter plot, understanding the scatter plot results, using the heatmap and interacting with the annotations. All of these subtasks were ranked highly for ease of use and understanding. The heatmap function was also listed as the most helpful function for accomplishing tasks.

Table 2. Task two result averages, ranked from 1 to 7. 1 being the least helpful/intuitive and 7 being the most helpful/intuitive.

How intuitive was the scatter plot and understanding the results.	5.875
How helpful was the annotation overlay in examining results.	5.625
How helpful was the heatmap in locating results.	5.75

5.3 Task 3

In task 3 we asked participants to use their prior selection from task 1 to narrow down the search results to find buildings of similar size with the most similarity to their selection. We then asked participants to locate the top three similar features for us on the map using the detailed results table. This final task built upon the two previous tasks by first requiring navigating and selecting a region of interest. Then, from task 2, the participant analyzed the overall results of their selection in both the scatter plot and spatially.

We started by telling participants that we wanted to find features within a specific size range with the most similarity. This required the participants to interact with the scatter plot results by filtering out all the results below and above certain values, in this case to find features of a certain size and with the most similarity.

Once an interaction occurred in the scatter plot, it is updated to the detailed results table that sorts all the results based on their similarity metric. Participants then used this table to select the top three features that were most similar. Using the detailed view they were able to then locate and point out where these features were on the LIDAR patches.

Table 3. Task three result averages, ranked from 1 to 7.

How easy was it to locate buildings of similar size	6.0
How helpful is the detailed list of ranked results.	5.625
How easy was it to locate a result on the map.	6.25

5.4 Results

We will start first by discussing the shortcomings. During our research we very quickly discovered that there is no single method for solving our problem; it is multi-faceted. One of the key areas of our method is encoding values. Since our encoded values are based on a triangulated mesh any artifacts or errors produced by the triangulation will percolate into the analysis. If the values are too ambiguous and do not provide enough detail, then trying to do further analysis or extracting other features is extremely difficult. We discovered this in trying to extract a road network. The encoded values we were using were too ambiguous; we had to come up with another metric and therefore we chose to use a navigation mesh. We also recognize that our solution is not a complete solution for all possible use cases for analyzing LIDAR data, but we believe that our approach is flexible enough to incorporate new analysis techniques that can be embedded into the final encoded image.

Overall for all three tasks, we received strong positive rankings on the follow-up questions we asked. At the end of the questionnaire we asked three general questions about the overall system. The first question asked participants to rank the ease of use for the application. The second question asked participants how similar they thought the

results were to what they had originally selected. Last, we provided an open-ended question asking participants what they thought was the most helpful feature. We received high ranks from our participants for the overall ease of use in using the interface, carrying out the tasks, and understanding the analysis results. The participants also thought they results they found were very similar to what they had originally selected.

Table 4. Final question result averages, ranked from 1 to 7. 1 being the worst and 7 being the best.

How would you rate the overall ease of use of the interface.	6.0
How similar do you think the results are from a selection.	5.625
What was the most helpful function.	Heatmap

6 Future Work and Conclusion

We explored the possibilities of using machine learning for classification and matching, but ultimately that required an extensive training set of data. The goal of our effort was to find a scalable, unsupervised approach that leveraged human analysis in an interactive tool for large data sets. But, for future work we see potential in machine learning to utilize selections made by someone using the application. As the system is used it can learn more about what a particular user is looking for and continually build up a repository of knowledge that can be applied to later analysis or future predictions. Machine learning would also be a good tie in with image analysis and our image based data.

Due to the nature of LIDAR it can grow exponentially when examining larger spatial regions and different points in time. The size of LIDAR datasets keeps growing. Sensors are collecting more and more data. The files can be hard to manage; even loading and viewing them can be a problem. It can be time consuming collecting different sources of data, processing them with a myriad of different tools, and converting LIDAR coordinate systems.

In this paper, we present and discuss a concept and the implementation of a system that is capable of analyzing and visualizing LIDAR data in real-time. We also tackled three major issues: scalability, interaction, and the ability to search. LIDAR data by nature can be massive, hard to manage, and visualize. We utilized GPU computing and image analysis to achieve our goal of real-time interaction and analysis. As part of our GPU processing step we independantly analyze each patch of LIDAR data to encode values into a 2D image.

We implemented our concept into an interactive visualization tool that allows users to easily and efficiently explore, find areas of interest, and analyze those areas. We incorporated several interactive tools for managing and organizing the data. Using the tool we ran an informal case study evaluation our system with 10 participants. Each participant performed a series of tasks that examined how easy the system was to use in navigating, exploring, and selecting areas of interest. Next we had each participant analyze the results of their selection through the use of a scatter plot and spatial analysis. In the last task we had participants isolate the data looking for specific features. Overall the participants for each tasked ranked the system very well. They found it responsive, easy to use and understand. They were also confident in the ability of the system to return similar results. Through the results and efforts of our work we believe we have accomplished what we set out to originally do.

Acknowledgments

This work was supported in part by the U.S. Department of Homeland Securitys VACCINE Center under award no. 2009-ST-061-CI0002.

References

[1] A. Akaydın and U. Güdükbay. Adaptive Grids: An Image-based Approach to Generate Navigation Meshes. *Optical Engineering*, 52(2):Article No. 027002, 12 pages, February 2013.

[2] T. Blaschke. Object based image analysis for remote sensing. {*ISPRS*} *Journal of Photogrammetry and Remote Sensing*, 65(1):2 – 16, 2010.

[3] G. Bradski. *Dr. Dobb's Journal of Software Tools*.

[4] T. Butkiewicz, R. Chang, Z. Wartell, and W. Ribarsky. Visual analysis and semantic exploration of urban lidar change detection. In *Proceedings of the 10th Joint Eurographics / IEEE - VGTC Conference on Visualization*, EuroVis'08, pages 903–910, Aire-la-Ville, Switzerland, Switzerland, 2008. Eurographics Association.

[5] R. Chang, T. Butkiewicz, C. Ziemkiewicz, Z. Wartell, N. S. Pollard, and W. Ribarsky. Legible simplification of textured urban models. *IEEE Computer Graphics and Applications*, 28(3):27–36, 2008.

[6] C. Clasen, F. A. Kruse, and A. Kim. Analysis of lidar data for emergency management and disaster response. In *Imaging and Applied Optics Technical Papers*, page RTu2E.2. Optical Society of America, 2012.

[7] T. Eaglin, X. Wang, W. Ribarsky, and W. Tolone. Ensemble visual analysis architecture with high mobility for large-scale critical infrastructure simulations, 2015.

[8] J. Estornell, L. A. Ruiz, B. Velzquez-Mart, and T. Hermosilla. Analysis of the factors affecting lidar dtm accuracy in a steep shrub area. *Int. J. Digital Earth*, 4(6):521–538, 2011.

[9] E. J. Hastings, J. Mesit, and R. K. Guha. Optimization of large-scale, real-time simulations by spatial hashing.

[10] T. Hermosilla, L. A. Ruiz, J. A. Recio, and J. Estornell. Evaluation of automatic building detection approaches combining high resolution images and lidar data. *Remote Sensing*, 3(6):1188–1210, 2011.

[11] M. Kallmann and M. Kapadia. Navigation meshes and real-time dynamic planning for virtual worlds. In *ACM SIGGRAPH 2014 Courses*, SIGGRAPH '14, pages 3:1–3:81, New York, NY, USA, 2014. ACM.

[12] M.-P. Kwan and D. M. Ransberger. Lidar assisted emergency response: Detection of transport network obstructions caused by major disasters. *Computers, Environment and Urban Systems*, 34(3):179 – 188, 2010.

[13] R. Richter and J. Dllner. Concepts and techniques for integration, analysis and visualization of massive 3d point clouds. *Computers, Environment and Urban Systems*, 45(0):114 – 124, 2014.

[14] J. C. Roberts. State of the art: Coordinated and multiple views in exploratory visualization. In *Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization*, pages 61–71, 2007.

[15] R. Scheepens, N. Willems, H. van de Wetering, G. Andrienko, N. Andrienko, and J. van Wijk. Composite density maps for multivariate trajectories. *Visualization and Computer Graphics, IEEE Transactions on*, 17(12):2518–2527, Dec 2011.

[16] M. Sedlmair, M. Meyer, and T. Munzner. Design Study Methodology: Reflections from the Trenches and the Stacks. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 18(12):2431–2440, 2012.

[17] K. Wagstaff, C. Cardie, S. Rogers, and S. Schroedl. Constrained k-means clustering with background knowledge. In *In ICML*, pages 577–584. Morgan Kaufmann, 2001.

[18] A. Zerger and D. I. Smith. Impediments to using {GIS} for real-time disaster decision support. *Computers, Environment and Urban Systems*, 27(2):123 – 141, 2003.

VASA: Interactive Computational Steering of Large Asynchronous Simulation Pipelines for Societal Infrastructure

Sungahn Ko, Jieqiong Zhao, Jing Xia, *Student Member, IEEE*, Shehzad Afzal, Xiaoyu Wang, *Member, IEEE*, Greg Abram, Niklas Elmqvist, *Senior Member, IEEE*, Len Kne, David Van Riper, Kelly Gaither, Shaun Kennedy, William Tolone, William Ribarsky, David S. Ebert, *Fellow, IEEE*

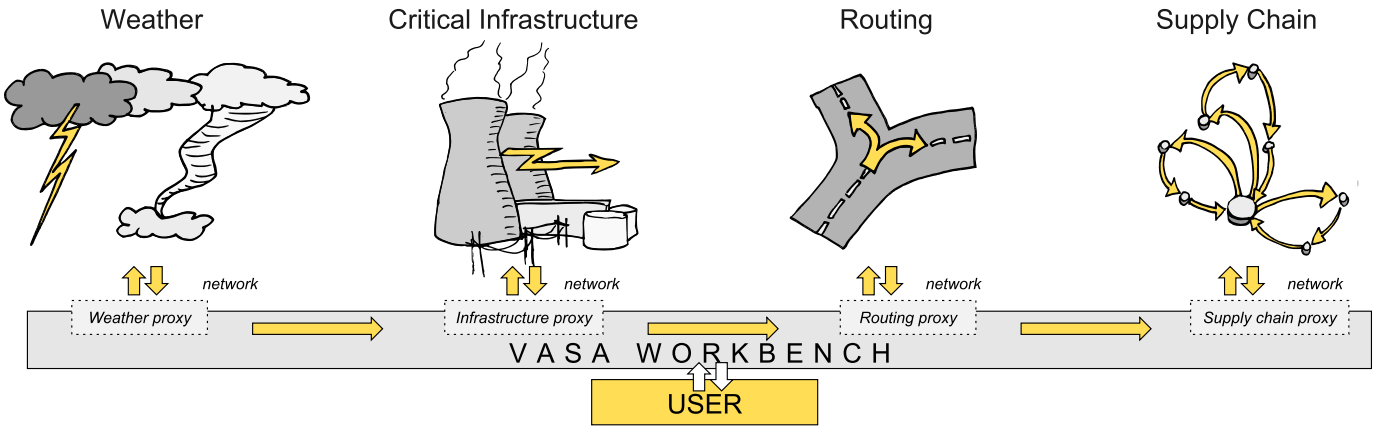


Fig. 1. Conceptual overview of the VASA system, including four simulation components for weather, critical infrastructure, road network routing, and supply chains, as well as the VASA Workbench binding them together.

Abstract—We present VASA, a visual analytics platform consisting of a desktop application, a component model, and a suite of distributed simulation components for modeling the impact of societal threats such as weather, food contamination, and traffic on critical infrastructure such as supply chains, road networks, and power grids. Each component encapsulates a high-fidelity simulation model that together form an asynchronous simulation pipeline: a system of systems of individual simulations with a common data and parameter exchange format. At the heart of VASA is the Workbench, a visual analytics application providing three distinct features: (1) low-fidelity approximations of the distributed simulation components using local simulation proxies to enable analysts to interactively configure a simulation run; (2) computational steering mechanisms to manage the execution of individual simulation components; and (3) spatiotemporal and interactive methods to explore the combined results of a simulation run. We showcase the utility of the platform using examples involving supply chains during a hurricane as well as food contamination in a fast food restaurant chain.

Index Terms—Computational steering, visual analytics, critical infrastructure, homeland security.

1 Introduction

Highways, interstates, and county roads; water mains, power grids, and telecom networks; offices, restaurants, and grocery stores; sewage, landfills, and garbage disposal. All of these are critical components of our societal infrastructure that help run our world. However, the complex and potentially fragile interrelationships connecting these components also mean that this critical infrastructure is vulnerable to both natural and man-made threats: twisters, hurricanes, and flash floods; traffic, road blocks, and pile-up collisions; disease, food poisoning, and major pandemics; crime, riots, and terrorist attacks. How can a modern society protect its critical infrastructure against such a diverse range of threats? How can we design for resilience and preparedness when perturbation in one seemingly minor aspect of our infrastructure may have vast and far-reaching impacts across society as a whole?

Simulation, where a real-world process is modeled and studied over time, has long been a standard tool for analysts and policymakers to answer these very questions (e.g., applications for modeling the real world [10]). Using complex simulations of critical infrastructure components, expert users have been able to create "what-if" scenarios, calculate the impact of a threat depending on its severity, and—last but not least—study optimal mitigation measures to address them. In fact, analysts have gone so far as to name "simulation as the new innovation" [35]: instead of endeavoring to produce the perfect solution once and for all, this new school of thought is to create a whole range of possible solutions and determine the optimal one using modeling and simulation. For example, during the Obama reelection campaign, it was reported that Organizing for Action data analysts ran a total of 62,000 simulations to determine voter behavior based on data from social media, political advertisements, and polling [43]. Basically, the philosophy with big data analytics driven by simulation is not to get the answer perfectly right, but to be less wrong over time [34]. Put differently, while it would be inappropriate to state—as others have done [2]—that big data will never somehow overtake theory, it is clear

- *Sungahn Ko, Jieqiong Zhao, Shehzad Afzal, Niklas Elmqvist, and David S. Ebert are with Purdue University in West Lafayette, IN, USA. E-mail: {ko, zhao413, safzal, elm, ebertd}@purdue.edu.*
- *Jing Xia is with Zhejiang University in Hangzhou, China. E-mail: xiajing@zjucadcg.cn.*
- *Xiaoyu Wang, William Tolone, and William Ribarsky are with University of North Carolina at Charlotte in Charlotte, NC, USA. E-mail: {xiaoyu.wang, ribarsky}@uncc.edu.*
- *David Van Riper, Len Kne and Shaun Kennedy are with University of Minnesota in Minneapolis, MN, USA. E-mail: {vanriper, lenkne, kenne108}@umn.edu.*
- *Greg Abram and Kelly Gaither are with University of Texas at Austin in Austin, TX, USA. E-mail: {gda, kelly}@tacc.utexas.edu*

Submitted to IEEE VAST 2014. Do not redistribute.

that large-scale simulation is a new and powerful tool in our arsenal for making sense of the world we live in.

Applying simulation to the scope of entire critical infrastructures—such as transportation, supply chains, and power grids—as well as the factors impacting them—such as weather, traffic, and man-made threats—requires constructing large *asynchronous simulation pipelines*, where the output of one or more simulation models becomes the input for one or more other simulations arranged in a sequence with feedback. Such a *system-of-systems* [12, 30] (SoS) will enable leveraging existing high-fidelity simulation models without having to create new ones from scratch. However, this approach is still plagued by several major challenges that all arise from the complexity of chaining together multiple simulations in this way: (C1) *monolithic simulations* that are designed to be used in isolation, (C2) *complex configurations* for each model, (C3) *non-standard data exchange* for passing data between them, and (C4) *long execution times* for each individual simulation that are not amenable to interactive visual analytics.

To address these challenges, we present **VASA** (Visual Analytics for Simulation-based Action), a visual analytics platform for interactive decision making and computational steering of these types of large-scale simulation pipelines based on a visual analytics approach. The VASA Workbench application itself is an interactive desktop application that binds together a configurable pipeline of distributed simulation components. It eables the analyst to visually integrate, explore, and compare the inter-related and cascading effects of systems of systems components and potential final alternative outcomes. This is acheive by visualizing both intermediate and final results from the simulation components using a main spatiotemporal view as well as multiple secondary views. The tool provides an interface for the analyst to navigate in time, including stepping backwards and forwards, playing back an event sequence, jumping to a particular point in time, adding events and threats to the timeline, and initiating mitigation measures. Moreover, it allows them to select between or combine different ensemble outputs from one simulation to be fed to other SoS components and explore consequences. Using this interface, an analyst could for example add a weather event (e.g., either an existing hurricane from a historical database, the union of several ensemble output paths, or simulation of a new one) to a particular time, and then step forward a week to see its impact on roads, the power grid, food distribution, and total economic impact in southern United States.

The simulation components provide the main functionality to the VASA platform. Each simulation component communicates with the Workbench using a representational state transfer (REST) API that standardizes the data and parameter exchange. The data flows and parameters passed in the pipeline can be configured using the Workbench application using a graphical interface. Furthermore, the Workbench also includes a local *simulation proxy* for each remote simulation component that provides real-time approximations of each simulation model to enable using them for interactive visual discourse. This feature also provides the computational steering functionality of the Workbench: after configuring a simulation run in an interactive fashion, the analyst can launch the (possibly lengthy) execution from the Workbench. The Workbench then provides tools to manage the simulation pipeline, for example to prematurely shut down a simulation component to accept a partial result, or to skip a particular run.

Our work on the VASA project has been driven by stakeholders interested in supply chain management of food systems, with an initial working example of a food production to restaurant system. For this reason, other than the VASA Workbench application and the protocols and interfaces making up the platform, we have also created VASA components for simulating weather (including storms, hurricanes, and flooding), the power grid, supply chains, transportation, and food poisoning. We describe these individual components and then present an example of how the VASA platform can be used to explore a what-if scenario involving a major hurricane sweeping North Carolina and knocking out a large portion of the road networks and power grid. We also illustrate how the tool can be used to simulate food contamination outbreaks and how this information can be used to track back the contaminated products to the original distribution centers.

2 Background

Visual analytics [38], can be a powerful mechanism to harness simulation for understanding the world. Below we review the literature in visual analytics for simulation and computational steering, as well as appropriate visual representations for such spatiotemporal data.

2.1 Simulation Models

The potential for applying visual analytics to simulation involves not only efficiently presenting the results of a simulation to the analyst, but also building and validating large-scale and complex simulation models. For example, Matkovic et al. [27, 28] show that visual analytics can reduce the number of simulation runs by enabling users to concentrate on interesting aspects of the data. Maciejewski et al. [23] apply visual analytics techniques to support exploration of spatiotemporal models with kernel density estimation and cumulative summation. This work was extended to a visual environment for epidemic modeling and decision impact evaluation [1]. Similarly, Andrienko et al. [5] propose a comprehensive visual analytics environment that includes interactive visual interfaces for modeling libraries and supports selection, adjustment, and evaluation of such modeling methods. Our work is different from this prior art in that our approach combines multiple components in a simulation pipeline, where each stage in the pipeline produces visualization for analysis.

Supply chain management is also a multi-decisional context where what-if analyses are often conducted to capture provenance and processes of supplies. Simulation is recognized as a great benefit to improve supply chain management, providing analysis and evaluation of operational decisions in the supply process in advance [37]. With the IBM Supply Chain Simulator (SCS) [9] and enterprise resource planning (ERP), IBM is able to visualize and optimize nodes as well as relations in the supply chain [20]. Perez also developed a supply chain model snapshot [31] with Tableau. However, existing visualizations of supply chain are mostly limited to either local supply nodes or a metric model rather than managing the overall supply process.

2.2 Computational Steering

Computational steering refers to provoding user control over running computations, such as simulations. Mulder et al. [29] classify uses of computational steering as model exploration, algorithm experimentation, and performance optimization. Applications include computational fluid dynamics (CFD) [13], program and resource steering systems [40], and high performance computing (HPC) platforms [7].

For all of the above applications, the user interface is a crucial component that interprets user manipulation for reconfiguration of data, algorithms, and parameters. Controlling, configuring, and visualizing such computational steering mechanisms is an active research area. Waser et al. proposed World Lines [41], Nodes on Ropes [42], and Visdom [33] as well as an integrated steering environment [33] to help users to manage *ensemble simulations*—multiple runs of the same or related simulation models with slightly perturbed inputs—of complex scenarios such as flood simulations. In the business domain, Broeksema et al. [8] propose the Decision Exploration Lab (DEL) to help users explore decisions generated from combined textual and visual analysis of decision models rooted in articifical intelligence.

2.3 Spatiotemporal Data

Spatiotemporal visual analytics systems enable users to investigate data features over time using a visual display based on geographic maps [3]. In these systems, color, position, and glyphs display features of different regions by directly overlaying the data on the map.

Many approaches to visual analytics for spatiotemporal data exist. Inspired partly by a survey by Anselin [6], we review the most relevant ones below. Andrienko and Andrienko [4] use value flow maps to visualize variations in spatiotemporal datasets by drawing silhouette graphs on the map to represent the temporal aspect of a data variable. Hadlak et al. [16] visualize attributed hierarchical structures that change over time in a geospatial context. Fuchs and Schumann [15] integrate ThemeRiver [17] and TimeWheel [39] into a map to visualize spatiotemporal data. Ho et al. [18] present a geovisual analytics

framework for large spatiotemporal and multivariate statistical flow data analysis using bidirectional flow arrows coordinated and linked with a choropleth map, histogram, or parallel coordinates plot. Our approach is different from those in that our system provide a visual analytics environment for managing and analyzing the results from multiple types of simulations.

Some approaches enable analysis of spatially-distributed incident data. Maciejewski et al. propose a system for visualizing syndromic hotspots [24, 22] while Malik et al. [25] develop a visualization toolkit utilizing KDE (Kernel Density Estimation) to help police better analyze the geo-coded crime data. The latter system is extended to a visualization system [26]where historic response operations and assessment of potential risks in the maritime environment can be analyzed. In our work we also employ KDE for visualizing spatial distrubution of ill people who consumed contaminated food in a supply chain.

3 Design Space: Steering System-of-System Simulations for Modeling Societal Infrastructure

Computational steering is defined as user intervention in an autonomous process to change its outcome. This approach is commonly utilized in visual analytics [38] to introduce a human analyst into the computation loop for the purpose of creating synergies between the analyst and computational methods. In our work, the autonomous processes we are studying are simulation models (often based on discrete event models) that are chained together into asynchronous simulation pipelines where the output of one or several simulations becomes the input to one or several other simulations. Such a simulation pipeline is also a *system-of-systems* [12, 30] (SoS): multiple heterogeneous systems that are combined into a unified, more complex system whose sum is greater than its constituent parts. Synthesizing all these components yields the concept of visual analytics for *steering system-of-system simulations*: the use of visual interfaces to guide composite simulation pipelines for supporting sensemaking and decisionmaking. In this work, we apply this idea to modeling societal infrastructure, such as transportation, power, computer networks, and supply chains.

In this section, we explore the design space of this concept, including problem domains, users, tasks, and challenges. We then derive preliminary guidelines for designing methods supporting the concept.

3.1 Domain Analysis

A wide array of problem domains may be interested in creating large-scale system-of-system simulation pipelines for studying impacts on societal infrastructure. Our particular domain is for business intelligence for supply chain logistics in the fast-food business, but we see multiple potential applications (each with a specific example):

- **Supply chain logistics:** Impact of large-scale weather events on the distribution of goods (particularly perishables, e.g., food).
- **Public safety:** Crime, riots, and terrorist attacks on critical infrastructure, such as on roads, bridges, or the power grid.
- **Food safety:** Incidence, spread, and causes of food contamination, often due to weather (power outage) or transport delays.
- **Cybersecurity:** Societal impact of cybersecurity attacks, such as on power stations, phone switches, and data centers.

3.2 User Analysis

The intended audience for computational steering of simulation models using visual analytics are what we call "casual experts": users with deep expertise in a particular application domain, such as transportation, supply chain, or homeland security, but with limited knowledge of simulation, data analysis, and statistics. Their specific background depends on the problem domain; for example, they may be business or logistics analysts for supply chain applications, police officers for public safety, and homeland security officials for food safety and cybersecurity. Because of this "casual" approach—a term we borrow from Pousman et al.'s work on casual information visualization [32]—our intended users are motivated by solving concrete problems in their application domain, but are not necessarily interested in configuring complex simulation models and navigating massive simulation results.

Even if our primary user audience is these casual experts, it is very likely that the outcome of a simulation steering analysis will be disseminated to managers, stakeholders, or even the general public [38]. Thus, a secondary user group for consuming our analysis products is laypersons with an even more limited knowledge in mathematics, statistics, and data graphics.

3.3 Task Analysis

Based on our review of the literature (Section 2) as well as feedback from domain experts, we identify a preliminary list of high-level tasks for steering system-of-system simulations for societal infrastructure:

- Increasing *preparedness* for potential scenarios;
- Improving the *resilience* of an organization; and
- Planning for *mitigation and response* to a situation.

3.4 Challenges

Modeling the real world is a tremendously difficult and error-prone process. However, we leave concerns about the fidelity, accuracy, and quality of a simulation to research within the simulation design. Rather, in this subsection we concern ourselves with the challenges intrinsic to connecting multiple individual simulation models into large-scale pipelines. In the context of simulation steering for such pipelines, we identify the following main challenges:

C1 **Monolithic simulations:** While individual high-fidelity simulation models exist for all of the above components and threats, these models are monolithic and not designed to work together.

C2 **Complex relationships:** Each high-fidelity simulation model consists of a plethora of parameters and controls that require expertise and training, which is exacerbated when several such models are combined into a single model.

C3 **Non-standard data:** No standardized data exchange formats exist for passing the output of one simulation model, such as for weather, as input to another model, such as supply chain routing.

C4 **Long execution times:** Most state-of-the-art, high-fidelity simulation models require a non-trivial execution time, often on the order of minutes, if not hours. Such time frames are not amenable for real-time updates and interactive exploration.

C5 **Uncertainty and fidelity:** Chaining together multiple simulations into a pipeline may yield systematically increasing errors as uncertain output from one model is used as input to another. This is compounded by the fact that heterogeneous simulation models may have different levels of fidelity and accuracy.

3.5 Design Guidelines

Based on our review of the problem domain, users, and tasks above, as well as the challenges that these generate, we formulate the following tentative guidelines for designing visual analytics methods for steering system-of-system simulation pipelines:

G1 *Simulations as standardized network services:* Distributing simulation models as network services avoids the trouble of integrating a monolithic design with another system (C1) and automatically provides a data exchange format (C3). The simulations also become decoupled, which means they can be parallelized and/or distributed in the cloud to manage long execution times (C4).

G2 *Simulation proxies for interactive response:* Meaningful sensemaking in pursuit of one of the high-level tasks in Section 3.3 requires real-time response to all interactive queries. This means that long execution times (C4) of simulation models in the pipeline should be hidden from the user. We propose the concept of a *simulation proxy* as an approximation of a remote simulation service that is local and capable of providing real-time response at the cost of reduced (often significantly) accuracy.

G3 *Visual and configurable relationships:* The interactive visual interfaces routinely employed in visual analytics may help to simplify and expose the complex configurations necessary for many high-fidelity simulation models (C2), even for non-expert users.

G4 *Partial and interruptible computational steering:* Once an analyst has configured a simulation run using simulation proxies (G2) and visual mappings (G3), the full simulation pipeline must be invoked to calculate an accurate result. A full-fledged simulation run may take minutes, sometimes hours, to complete. The computational steering mechanisms provided by the software should provide methods for continually returning partial results [14] as well as interrupting a run halfway through.

G5 *Visual representations of both intermediate and final results:* To fully leverage the power of visual analytics, we suggest using interactive visual representations of simulation results. Such visualizations should be used for both intermediate data generated by a simulation component anywhere in the pipeline—which would support partial results and interrupting a run at any time—as well as for the final results. All visual representations should be designed with uncertainty in mind (C6), and providing intermediate visualizations should also help in exposing propagation of increasing error. Finally, it may also be useful to use visual representations for the approximations created by simulation proxies (G2), but these should be clearly indicated as such.

4 VASA: Overview

As previously described, our VASA system is a distributed component-based framework for steering system-of-system simulations for societal infrastructure. Figure 1 gives a conceptual model of the system architecture. At the center of the system is the VASA Workbench (Figure 2), a user-driven desktop tool for configuring, steering, and exploring simulation models, impacts, and courses of action. The workbench provides a visual analytics dashboard based on multiple coordinated views, an event configuration view, and a computational steering view. The workflow of the workbench revolves around initiating, controlling, analyzing, exploring, and handling events from the remote simulation components as well as the local simulation proxies.

Within the dashboard, events are displayed in a selectable calendar view (a) where each event's name, dates and a user-selected representative attribute (e.g., storm's maximum wind speed) are shown. The selected events from (a) are listed based chronologically in the event viewer (b) where a user can select times for investigation. In (b-1), various options are provided, including initiating simulations (e.g., cyberattack, storm simulations, distribution re-routing), selecting combinations of events (union, intersection, difference), selecting event visualization modes (polygons, contours), and chronological playback.

Users can fix a time within an event for comparison (right-clicking on a event's black rectangle) and a red mark is shown in the upper right corner of the associated rectangle(b-2), and the impact is shown in the main geospatial view (d-1). We provide a legend window (c) for selected properties (e.g., distribution centers, restaurants, power plants and other infrastructures) and the geographical view (d) provides the simulation results including event evolution, routing paths, and impacts on critical infrastructures. A food delivery schedule to each store within a supply chain is provided in (e) where the x-axis presents corresponds to different restaurants while the y-axis represents different food processing centers or different types of foods. Here, the darker the red, the larger the quantity of the delivered food. The quantity information is provided in a tooltip that helps a user to estimate possible losses. This view enables traceback analysis (e.g., which type of food was contaminated from which processing centers, how much contaminated food was delivered to which store) for food contamination incidents.

5 VASA: Components

Our current VASA suite consists of four simulation components that implement the VASA interface: components for weather, critical infrastructure, routing, and supply chains. We review each of these next.

5.1 Weather Component

In order to provide clients with a one-stop source for weather data, we implement a server that asynchronously amasses data from various online sources and presents it to clients through a RESTful web interface. This provides access to various data through a singly authenticated service that provides consistent and convenient APIs for data acquired from many sources.

5.1.1 Simulation Model

For example, a collaboration of several research centers runs the ADCIRC model during hurricane season off the east and gulf coasts of the U.S. When storms are present, these models are run every four hours, producing ADCIRC-formatted datasets at fixed intervals forward from the initial times. These results are made publicly available using THREDDS and OPeNDAP for cataloging, discovery and data access. When this data appears, we import it onto a VASA server, and provide a simple RESTful API to access the data in convenient multiresolution formats. Similarly, NOAA produces wind-speed probabilities along the tracks of storms as contours at 34, 50, and 64-knot levels. This data is also imported asynchronously onto the VASA service and provided through the VASA RESTful API.

5.1.2 Simulation Proxy

The proxy in this component has two roles. The first role is to prepare all event data sets from the remote event server. Therefore, the system first checks for new updates from the server. If there is a new update, it retrieves the data and saves it on the local workbench for faster loading. The second role is to visualize new status of an event on the date that a user selected and notify the status change of the event to other proxies. An example status change is a user changing the start date of a hurricane in the event viewer. When this happens, the proxy visualizes a new status of the hurricane on the date and notifies this change to other components, which initiates each proxy's work (e.g., estimating an area without power and impassable roads).

A user can select the hurricane visualization type either as polygons or contours for estimation by clicking a button as shown in Figure 2 (b-2, the last button). In the polygon mode, two probability models (blue with two different opacities) are projected as shown in the magnification view in Figure 2. Here, the smaller polygon means an expected path with high probability, and a larger one presents an expected path with low probability. When a user fixes a hurricane, the hurricane turns red for comparison to other paths (of other hurricanes). For example, in Figure 2 the path of Hurricane Irene on August 24, 2011 is projected (blue) and the path of Hurricane Sandy in October 27, 2012 is presented in red for comparison.

In the contour mode, hurricanes are drawn using three different sizes of contours, each of which represents mean areas in different wind speeds (e.g., Hurricane Irene in our simulation model has 64 knot highest wind speed at the innermost contour, and 34 knot lowest wind speed at the outermost contour as shown in Figure 6). To utilize different wind speeds in simulation steering, a user can set up a threshold for infrastructures (e.g., a power generation unit is disabled if the wind hitting the plant has speed higher than 34 knot). In addition, a user can apply one of the contours for a time. For example, Figure 6 (top-right) presents which power generation units are affected when a contour with 34 knot hits the area. Here red circles represent affected restaurants and red circles present the impacted power generation units supplying electricity to those restaurants.

5.1.3 Implementation Notes and Performance

From the client's point of view, the VASA API consists of URLs that encode procedures and parameters that, when issued, return JSON objects containing the results. This provides a very simple interface for use both by browser-based visualization UIs that use AJAX to issue requests asynchronously, and other native platforms that provide equivalent access through language-specific interfaces.

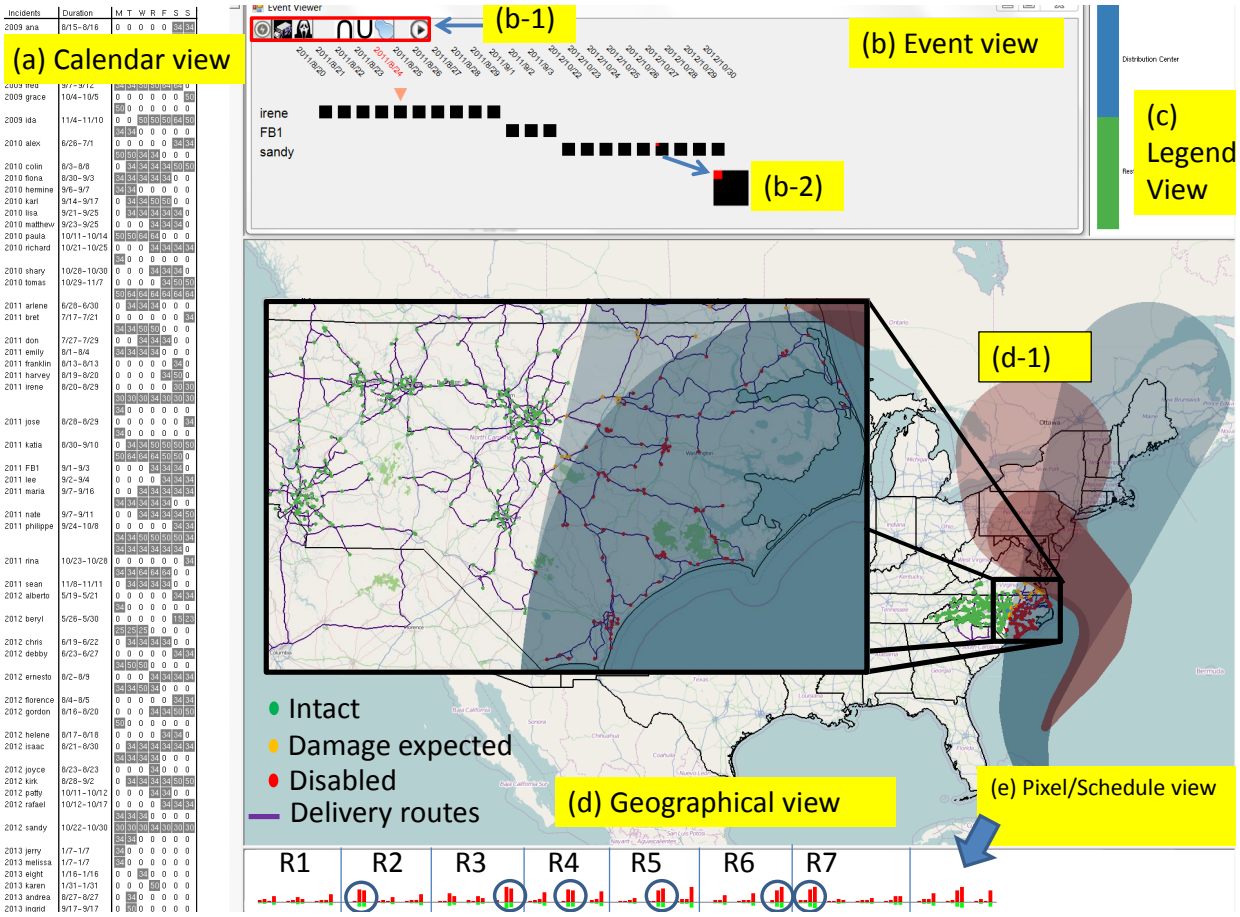


Fig. 2. Multiple coordinated views in the VASA Workbench. (a) Calendar view with available events (e.g., weather, food poisoning, cyberattack, etc). (b) Event timeline for configuring events. (b-1) Event buttons. (b-2) Fixed event. (c) Map legend. (d) Geographical map. (d-1) Hurricane (red). (e) Pixel/Schedule view showing food deliveries. Each area divided by a blue line means a route that visits 3–4 restaruns, 3 time a week. This view also can be used for pixel-based visualization.

5.2 Critical Infrastructure Component

Widespread emergencies such as hurricanes, flooding, or cyberattacks will often affect multiple societal infrastructures. High winds and flooding from a hurricane, for example, could knock out parts of the power grid, the effect of which would cascade to traffic signals, the communications network, the water system, and other infrastructures. The flooding might simultaneously make parts of the road network impassable. These breakdowns would affect critical facilities such as schools, hospitals, and government buildings. For longer-lived disasters, food distribution might break down due to power outage, route disruption, or other cascading effects. The purpose of VASA's critical infrastructure component is to simulate how such external emergencies, modeled in other components, will impact critical infrastructure.

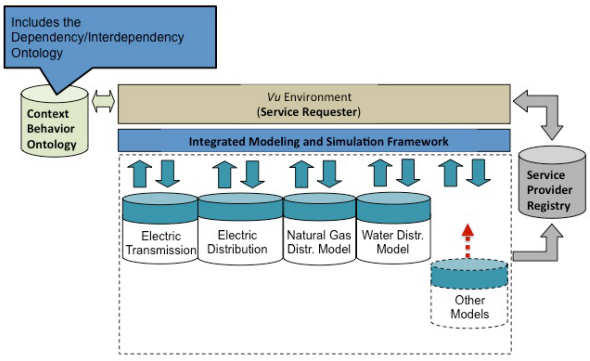


Fig. 3. Vu environment showing the modular structure where different simulation submodels can be inserted.

5.2.1 Simulation Model

To capture these complex, multifarious, and dynamic effects, we developed a simulation model that takes into account the interrelationships between critical infrastructure systems. The simulation is built within the Vu environment (Figure 3), which provides a rule-based framework for integrating multiple infrastructure components at a high level. This results in an interdependency ontology. Thus, for example, a breakdown of a power substation would immediately cascade to power loss at points on its distribution network. If a school were a node in the distribution network, it would be switched to backup power that, after a given time, would also shut down. Likewise, telecommunication nodes would switch to backup power that might also shut down after its prescribed duration. There could also be outages due to power load imbalances at other points in the grid.

These interlaced critical infrastructures are captured in a set of networks, with each node having a set of properties according to its category and the edges providing a dependency rule according to the category and state of the connected nodes. Relations between networks is captured by edges between nodes in the two networks. The timings of interdependencies and state changes are set according to a universal clock, so that any simulation of cascading effects evolves over time and space (since nodes are geographically located). The rules for networks and interdependencies are set in consultation with experts (in the case of the power grid, for example) or through consultation of the appropriate literature for an infrastructure. However, some of the interdependencies are not directly known, even by experts, since measures or simulations linking some infrastructures have never been done or validated. In this case, we define plausible rules that produce outcomes consistent with experience. This is in fact an advantage of the

Vu approach in that it permits investigating interdependencies at a high level with a quick adjustment of the model and turnaround of results. These interdependencies can then be studied in depth as needed.

The Vu approach will not capture detailed interdependency effects, transients, or complex interactions. However, it has been shown to be quite successful as a high-level simulator for a range of applications. To achieve this, it is modular, allowing new modules to be dropped in (Figure 3) and, as soon as the interdependency ontology is set up, simulations to be run. This makes it easy to configure simulations for several different infrastructures and even involving other types of simulation models, such as population and economic models.

5.2.2 Data

Our prototype system currently uses data from the state of North Carolina, and the data collection and organization process involves locating and identifying components of the various infrastructures for the state. We use publicly available data sources, in some cases identifying infrastructure components by indirect means. For example, comprehensive information about the electrical grid is closely held by the utility companies. However, we have shown our results to utility company officials and received confirmation as to their high level accuracy.

The infrastructures we currently model include the electric grid; the communications network including TV stations, radio stations, cellular switch controls, and cell towers; transportation facilities including airports, bus terminals, rail lines and terminals, bridges, tunnels, and ports; the road network including main and secondary roads; natural gas pipelines and pumping stations; critical facilities including fire stations, police stations, schools, hospitals, emergency care facilities, manufacturing locations, government buildings, and hazmat facilities.

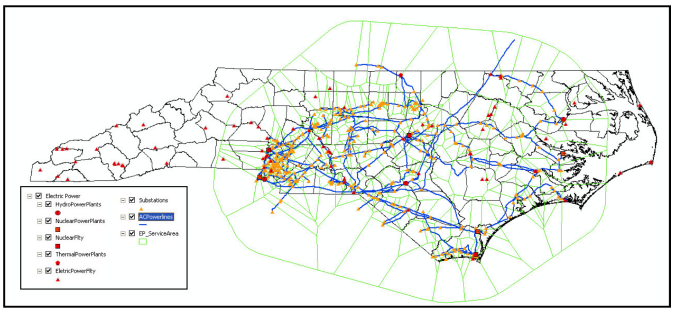


Fig. 4. Power transmission grid with parts of the distribution network.

Figure 4 shows the electric grid, which includes the complete transmission network down to substations for both North and South Carolina. Parts of the distribution network are also included, especially for critical installations. Figure 5 shows the transportation network for North Carolina, including roads, airports, rail lines, etc. For the purposes of VASA, we have also added store and distribution center locations for a large food chain in North and South Carolina. These facilities are linked to the power grid and road networks.

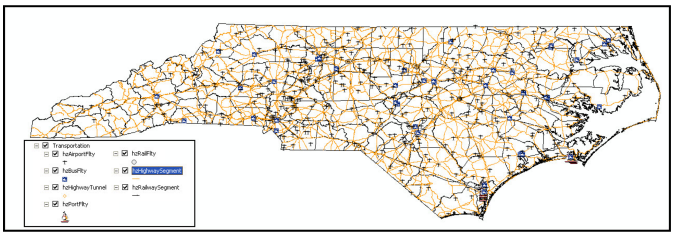


Fig. 5. Transportation network including transportation facilities.

5.2.3 Simulation Proxy

The proxy for the critical infrastructure component maintains a simplified connectivity network of critical infrastructure. In this graph, restaurants are connected to the nearest plant for impact approximation due to an event. When the proxy receives a signal of a new event (e.g., storm path change, new day for approximation), it computes which infrastructures are affected by the event. For example, when a user moves the hurricane simulation forward to a new day, our proxy checks which infrastructures are newly affected and produces an estimate and its corresponding visualization (e.g., condition color changes for restaurants affected by power plant disruptions).

5.2.4 Implementation Notes and Performance

As for all the other VASA simulation components, we use a web service that can accept requests from the VASA Workbench and send either complete or approximate simulation results ready to be presented in the user interface (see Figure 1). The critical infrastructure server itself has two components. One contains a searchable database of the pre-computed ensemble of simulation runs. The other accepts current storm path and other inputs from the weather server, converts them into courses of action, and computes a fresh set of cascading infrastructure disruption results. When a request is issued via the user interface, the Simulation Proxy determines the weather inputs to send to the ensemble database component that immediately selects the closest ensemble simulation for use in the visual analysis. This proxy is then replaced by the more accurate result based on the current weather simulation as soon as it is available. Therefore an emergency response manager can make initial decisions based on the proxy and then refine them, if desired, once the up-to-date result is available.

5.3 Supply Chain Component

Most food systems involve a number of firms from on-farm production of inputs through processing, distribution and retail sales. For the fast food system in VASA, three different firms have collaborated to provide the data on normal system performance: a vertically integrated poultry firm (hatchery to processed chicken), a warehouse and distribution firm, and a fast food restaurant firm. Each firm contributed data from their portion of the supply chain to enable modeling of product movement from farm to restaurant. The type of data provided includes geospatial information on the facilities involved (e.g., feed mills, hatcheries, poultry farms, poultry processing facilities, distribution centers and restaurants), normal transportation routes and scheduled times from each facility to the next facility in the system and details on actual shipment quantities on average (hatchery through processing) or actual shipment records for a limited time frame (distribution centers to restaurants). As an illustration of the amount of data that drives these systems, one week of data on product delivered from the two distribution centers to the nearly five hundred individual restaurants alone is over 120,000 individual records.

Hurricanes pose significant risks for normal supply chain operation from impassable roads, power outages and floodings disrupting facility operation and distribution of products throughout the system. Understanding which routes and locations are likely to be at risk from a storm would enable a firm to develop contingency plans in advance of a storm, thus reducing operational losses immediately after a storm. Given that daily sales at larger fast food restaurants can be $4,300-$7,400, losses can mount up quickly. If in the case of an impending storm or immediate aftermath, near real time rerouting could enable firms to most efficiently maintain their distribution systems for both maintaining product distribution and retrieving of food from restaurants without power to minimize spoilage losses.

5.3.1 Simulation Model

Food contamination can occur both intentionally or as a malicious act at any point in the supply chain and can result in significant public health consequences, from morbidity to mortality. While firms are required to have information one step forward and one step back in their supply chain, they often have difficulty gaining visibility beyond that. By gathering data from each step in the supply chain, it is possible to trace product from farm through to restaurant and from restaurant back to farm. Using data on actual lot sizes from the firms involved, two illustrative contamination scenarios were constructed to illustrate how differently seemingly similar contamination scenarios would transpire. This system also illustrated a common problem of “hidden nodes” in

the system, i.e., facilities that one firm in the system does not realize are part of its supply chain. One of the poultry slaughter and processing facilities ships raw poultry to a further processing facility that then ships the resulting product to the distribution centers. If there were a contamination at the "blind" facility, neither the distribution firm for the restaurant firm would initially know that it was part of their supply chain. A contamination scenario builder is now under development that would enable users to model a wide range of contamination events and see how they would propagate through the supply chain.

Our simulation model can generate food-borne illness data based on an approach similar to the Sydovat [21] system. There are two major components of the model for generating synthetic illness data: temporal and spatial data. A time series is contructed from its individual components (day-of-week, interannual, interseasonal, and remainder) similar to seasonal trend decomposition. To generate the time series of food-borne illnesses for a user-injected restaurant location, the user defines the mean daily count of illnesses along with seasonal and day of week components. If the historical food-borne illnesses data is available then seasonal and day of week components can be randomly selected from this historical data. Spatial locations for temporal data are generated based on the density distribition that approximate the population in that area. Additionally, users can customize the grid size and density distributions.

5.3.2 Simulation Proxy

Our simulation proxy for the supply chain component maintains a low-fidelity representation of the transport network. This is used together with the weather polygons to approximate when a distribution center and store must shut down. For food-poisoning data, this inherently contains spatially-distributed points of ill people simulated based on the simulation model (Section 5.3.1). To visualize the spatial distribution and the hotspots of the poisoned people, the proxy in this component uses a modified variable kernel density estimation technique with varying scales of the parameter of estimation based upon the distance from a patient location to the k_{th} nearest neighbor [36]. The model used for estimating the number of people poisoned is the same model utilized in Maciejewski et al. [1, 22], but we adjust parameters to consider different population densities in different regions.

5.3.3 Implementation Notes and Performance

The supply chain component is built in ArcGIS and Arc Network Modeler so that storm impacts can model solutions accounting for restaurants out of service (power, flooding) and impassable roadways.

5.4 Routing Component

The purpose of the routing component is to provide a mechanism for other VASA components to find appropriate routes from one facility to another given a dynamically changing world model, where roads may become impassable due to weather or other widespread emergencies.

5.4.1 Data Model

We obtained the addresses of two distribution centers and 505 fast-food outlets, as well as the route information that links the centers to the outlets. We geocoded the addresses using the Environmental Systems Research Institute (Esri) ArcGIS 10.2 Server with the Network Analyst extension, and StreetMap Premium for ArcGIS (TomTom North America data) Geodatabase. We then calculate N shortest path routes, where N is the number of routes specified in the input data, using Esri's Network Analyst Route tool and the StreetMap Premium road network. The road network has a long list of attributes used to determine the shortest route, including road class, speed limit, number of lanes, and weight restrictions.

5.4.2 Simulation Model

The input to the routing component is a GeoJSON polygon representing an area impacted by severe weather (such as a hurricane). The component ingests the GeoJSON object as a polygon barrier in the road network. Attributes of the road network are weighted to create a friction surface which iterates through routing options to determine the optimal route. The model does not currently include current traffic conditions or construction activity, but these factors could be added in the future. Each route minimizes the travel time between the distribution center and the first store or between stores. This set of routes represented the baseline scenario—how delivery trucks would travel under normal circumstances. Since delivery trucks can no longer reach outlets covered by the weather barrier, the routing service recomputes the routes with the barrier in place and returns new routes which avoid the outlets and roads covered by the barrier. If the barrier covers a distribution center, no deliveries will be made to outlets serviced by the center. The routes are output as a set of large GeoJSON objects and sent back to the caller.

5.4.3 Simulation Proxy

The main focus of the proxy in the routing component is on approximating the number of routes that will be replaced if a complete simulation result exists. The proxy investigates which nodes in routes are expected to be disabled when there is an event. Then, after the investigation, it builds a polygon by connecting outer-most nodes and visualizes the polygon. This gives awareness to a user that the routes in the polygon are likely to be changed after a complete simulation. A user can initiate the simulation by clicking the "run" button (Figure 9).

5.4.4 Implementation Notes and Performance

The goal was to use as much Commercial Off-The-Shelf (COTS) software as possible when implementing the routing model. The Esri suite of Geographical Information System (GIS) tools is widely used in a variety of industries and provides a robust set of tools and data. Specifically, we used ArcGIS Server 10.2 with the Network Analyst extension. The server provides web-based services through REST endpoints and provides a robust API accessed with HTTPS GET or POST requests. The VASA workbench initiates a request to the routing service by providing a GeoJSON representation of the affected area. The affected area polygon is input to Network Analyst Service to recalculate the route to traverse around the affected area. The response is two large GeoJSON objects containing a list of outlets no longer reachable, incremental travel time between stops, and the new route. Currently, the route processing requires 2-3 minutes to complete; this can be significantly improved when a production server is commissioned.

6 Examples

We showcase the utility of the VASA Workbench and our current simulation components using three examples: the impact of weather on macro-scale supply chains, foodborne illness contamination and spread, and a simplified cyber-attack on the power grid infrastructure.

6.1 Supply Chains in Hurricane Season

Our first example is the potential impact of hurricanes on North Carolina's critical infrastructure, especially our food distribution network, in North Carolina (NC). Our exploration begins by selecting appropriate historial hurricanes for examination using the calendar view as shown in Figure 2, where each hurricane name, duration, and selected summary attribute (e.g., maximum hurricane wind speed) are provided. While we investigate the paths of these historical hurricanes, we see that Irene in 2011 and Sandy in 2012 passed over NC. Because Sandy passed over only a small area in upper NC (Fig 2 (d), red hurricane polygon), we choose to focus on Irene for further investigation.

One interesting date is August 27, 2011 when Irene passed directly over eastern NC, an area with many power generation facilities, as shown in Figure 6 (top-right, purple circles). After we set up the wind tolerance value for these facilities to be 34 knot, our hurricane proxy instantly estimates which restaurants will be impacted based on the relationships between the units and the restaurants and colors the impacted restaurants red. Here, we also initiated a complete simulation for power outages and transportation network damage. Next, a polygon is shown representing an area where restaurants are disabled and which roads are blocked (bottom-left in Figure 6). To efficiently manage distribution, this impact requires the food provider to change its

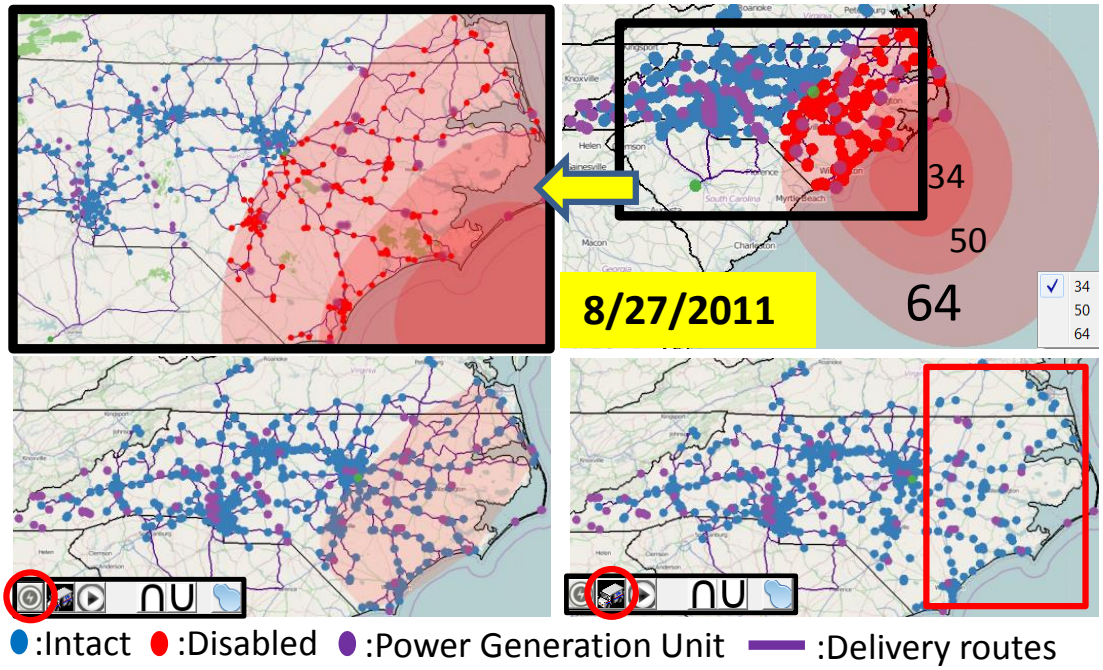


Fig. 6. In this simulation, power generation units were hit by up to 34 knot during Hurricane Irene on August 27, 2011. Our hurricane proxy instantly estimates the impacted restaurants (right-top, left-top). Note that one distribution center (green) is outside the hurricane. After a complete power-grid simulation run is finished (by clicking the circled lightning button), a polygon representing the power outage area is shown. Next, this polygon is sent for use in computing new food delivery paths. Note that food is not delivered to the power outage area (right-bottom, red box).

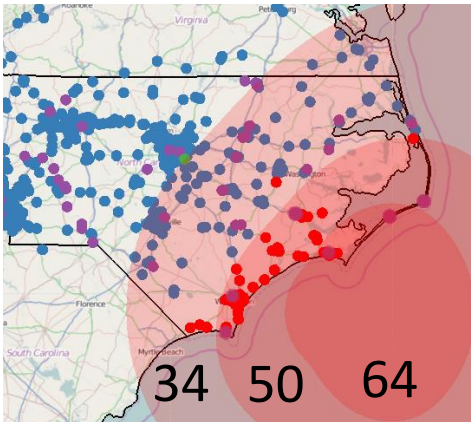


Fig. 7. If the power generation units could have resisted up to 50 knot wind, the number of impacted restaurants will be much smaller.

delivery schedule, and this new routing is computed based on the impacted restaurant polygon and road conditions (e.g., blocked by flooding). After a simulation to compute the new routes (by clicking the truck button in a red circle, right-bottom Figure 6), we see that the updated delivery paths do not include the affected restaurants. The economic loss caused by this event is estimated based on the model in Section 5.3 as being up to $1.13 million. Another possible what-if question is "How different would the result be if the power generation units can resist winds up to 50 knot?" Figure 7 shows the first step of the analysis where we see many fewer restaurants affected compared to Figure 6 top-right (units are resilient to 34 knots). In this case, the estimated losses are less than $333,000.

6.2 Fast Food Contamination

Food poisoning is an illness caused by eating contaminated food containing viruses, bacteria and germ-generated toxins. There are many possible causes of food contamination including storage at inappropriate temperatures [19], improper food handling, and cross-contamination during processing or packaging. As unfortunately experienced several times per year, tracing back the cause of the contamination is a very difficult lengthy process. In this example, we explore a hypothetical scenario demonstrating how VASA can be used to trace-back the root causes of an incident of foodborne illness.

To create the distribution of the ill population, we simulate the distribution of contaminated food to stores, then simulate the illnesses in the neighboring areas using the simulation model discussed in Section 5.3. This creates the common base scenario of reports of people who are ill, their date of illness and their location to create the food contamination scenario for the trace-back investigation.

For example purposes, we simulated these illnesses occurring during a three day span (September 1, 2011 to September 3, 2011) as shown in Figure 8. Since this is almost one week after Hurricane Irene, one may assume that power outages during the storm could be the possible reason behind the contamination. To confirm this hypothesis, we looked at the hot spots in Figure 8 and identified the stores closest to these hot spots. On cross comparison, we can identify the common products/lots in those stores, their distribution center, as well as their delivery mechanisms. As shown in Figure 8 bottom matrices, the rows represent 3 food processing centers and 4 types of food, and there is a column for each restaurant. Each cell is colored such that the darker the red color, the higher the amount of each product provided. Here, the restaurants in the affected area that are selected in the box in the top-left are highlighted with light green boxes. For stores S9 and S12, only one food processing center provided products, while other processing centers supplied most of the food throughout the network. Upon further inspection, one can determining that 3rd and 4th row product lots are common in most of the restaurants where individuals are. Some example routes are shown in Fig. 2 (e) where each route supplies 3-4 restaurants. A red bar means the supplied food and the green bar means the food consumed at a restaurant. Here, we see that a large amount of the third and fourth foods (blue circles in Fig. 2 (e)) are delivered and will all be consumed within a few days. Therefore, these two product lots are good candidates for further inspection in tracing back the contaminated food item.

6.3 Cyberattack on Critical Infrastructure

Part of the mission of the VASA project is to study the impact and mitigation of man-made attacks on societal infrastructure. Cybersecurity is becoming an increasingly important threat to modern society [11]

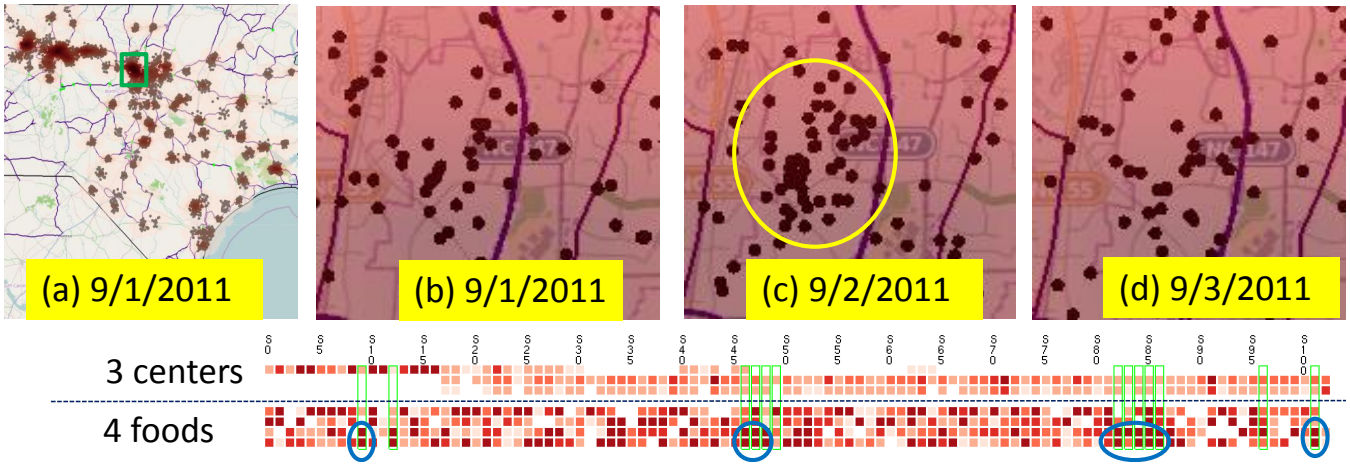


Fig. 8. Ill people caused by contaminated food is presented using a KDE hotspot visualization. In (a), the darker location has a larger number of poisoned people. Brown points mean ill people in the reported location. The locations highlighted by a green box in (a) is magnified in (b), (c) and (d) on different dates. As the timeline shows, the number of ill people increased until 9/2/2011, then started decreasing on 9/3/2011. The bottom matrices show which food processing centers (1–3) were involved and which foods (1–4) were delivered to which store in 8/30/2011, two days before the illness. Here, the restaurants in the light green boxes are the those selected by the thicker green box in (a). We see that a large quantity (darkest red pixels in blue circles) of two foods (third and fourth rows) are commonly provided to restaurants in the area.

Fig. 9. An example of the cyberattack simulation. (left) A user selects to disable power transmission by a cyberattack in the option menu and selects the region shown in the red rectangle. One power plant (purple dot) is included within this rectangle (shown in the dark red circle). (second-left) The infrastructure proxy instantly estimates the affected restaurants (red dots), and a full simulation is initiated. (second-right) Power outage regions are presented by the polygon, and new distribution routes are computed. (right) The new routes are shown as paths and do not include the affected restaurants but, unlike the hurricane scenario, all roads are available for food distribution. For comparison, see the path radiating from the polygon that was not allowed in the hurricane scenario.

and may have a significant effect on an increasingly connected society where power plants and substations are all controlled from afar.

While we do not yet provide a cyberattack module for VASA, many of the simulation components provide direct access to changing the state of particular infrastructure components through VASA. This enables us to simulate a cyberattack by, for example, shutting down a particular or several specific critical infrastructure component even if it is not affected by weather or other natural threats. Figure 9 shows a screenshot of an analyst studying precisely such a scenario where a power plant has been disabled by a cyberterrorist. Here the analyst simulates that the terrorist shuts down the power transmission grid for one of the two power plants in the town by drawing a red rectangle (left) around it. Then, the infrastructure proxy instantly estimates possible affected restaurants and a full simulation is initiated for more accuracy (second-left). The simulation result is presented by a polygon and new route computations can be initiated (second-right). The new routes with impacted restaurants are visualized. Note that this routing example is different from the hurricane case because roads are still passable: purple paths are still shown within the polygon (right).

7 Conclusion and Future Work

We have introduced the notion of visual analytics for simulation steering within the context of societal infrastructure. To our knowledge, ours is the first to study visual analytics for simulation from a *systems-of-systems* [12] perspective, where multiple heterogeneous—often physically distributed—systems are combined into a unified, more complex system in which the linkages between components provide a sum greater than its constituent parts. This notion transcends individual simulation models and instead chains together multiple high-fidelity simulations into large-scale asynchronous pipelines. The VASA system we presented as a practical example of such an approach is a distributed application framework consisting of a central Workbench controlled by an analyst and a set of loosely coupled simulation components implemented as distributed network services.

Big data simulation is a powerful new tool for data science, and while our work on applying visual analytics to this domain is conceptually complete, it really only scratches the surface of what is possible. Future work on the VASA system will involve integrating even more advanced and detailed simulation components, such as high-fidelity power grid models, gas pipelines, and power plants for energy infrastructure; bridges, tunnels, and causeways for transporation networks; and hospitals, police stations, and fire stations for societal infrastructure. In doing so, we envision designing additional novel visual representations and interactions for configuring these components as well as visualizing their proxy, intermediate, and final results.

Acknowledgments

This work was supported in part by the U.S. Department of Homeland Security's VACCINE Center under award no. 2009-ST-061-CI0002.

REFERENCES

[1] S. Afzal, R. Maciejewski, and D. S. Ebert. Visual analytics decision support environment for epidemic modeling and response evaluation. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 191–200, 2011.

[2] C. Anderson. The end of theory: The data deluge makes the scientific method obsolete. *Wired Magazine*, June 2008.

[3] N. Andrienko and G. Andrienko. *Exploratory analysis of spatial and temporal data: A systematic approach*. Springer Verlag, 2006.

[4] N. V. Andrienko and G. L. Andrienko. Interactive visual tools to explore spatio-temporal variation. In *Proceedings of the ACM Conference on Advanced Visual Interfaces*, pages 417–420, 2004.

[5] N. V. Andrienko and G. L. Andrienko. A visual analytics framework for spatio-temporal analysis and modelling. *Data Mining and Knowledge Discovery*, 27(1):55–83, 2013.

[6] L. Anselin. From SpaceStat to CyberGIS: Twenty years of spatial data analysis software. *International Regional Science Review*, 35(2):131–157, 2012.

[7] J. Biddiscombe, J. Soumagne, G. Oger, D. Guibert, and J.-G. Piccinali. Parallel computational steering and analysis for HPC applications using a ParaView interface and the HDF5 DSM virtual file driver. In *Proceedings of the Eurographics Conference on Parallel Graphics and Visualization*, pages 91–100, 2011.

[8] B. Broeksema, T. Baudel, A. G. Telea, and P. Crisafulli. Decision exploration lab: A visual analytics solution for decision management. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):1972–1981, 2013.

[9] S. Buckley and C. An. Supply chain simulation. In *Supply Chain Management on Demand*, pages 17–35. Springer, 2005.

[10] L. Costa, O. Oliveira, G. Travieso, F. Rodrigues, P. Boas, L. Antiqueira, M. Viana, and L. Rocha. Analyzing and modeling real-world phenomena with complex networks: a survey of applications. *Advances in Physics*, 3(60):319–412, 2011.

[11] R. Deibert. Towards a cyber security strategy for global civil society? Technical report, The Canada Centre for Global Security Studies, 2011.

[12] D. DeLaurentis and R. K. Callaway. A system-of-systems perspective for public policy decisions. *Review of Policy Research*, 21(6):829–837, 2004.

[13] H. Doleisch, H. Hauser, M. Gasser, and R. Kosara. Interactive focus+context analysis of large, time-dependent flow simulation data. *Simulation*, 82(12):851–865, 2006.

[14] D. Fisher, I. O. Popov, S. M. Drucker, and m. c. schraefel. Trust me, I'm partially right: incremental visualization lets analysts explore large datasets faster. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 1673–1682, 2012.

[15] G. Fuchs and H. Schumann. Visualizing abstract data on maps. In *Proceedings of the International Conference on Information Visualization*, pages 139–144, 2004.

[16] S. Hadlak, C. Tominski, H.-J. Schulz, and H. Schumann. Visualization of attributed hierarchical structures in a spatiotemporal context. *International Journal of Geographical Information Science*, 24(10):1497–1513, 2010.

[17] S. Havre, B. Hetzler, and L. Nowell. ThemeRiver: visualizing theme changes over time. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 115–124, 2000.

[18] Q. Ho, P. H. Nguyen, T. Åström, and M. Jern. Implementation of a flow map demonstrator for analyzing commuting and migration flow statistics data. *Procedia - Social and Behavioral Sciences*, 21:157–166, 2011.

[19] B. C. Hobbs. *Food poisoning and food hygiene*. Edward Arnold and Co., London, United Kingdom, 1953.

[20] A. Kamran and S. U. Haq. Visualizations and analytics for supply chains. Technical report, IBM, February 2013.

[21] R. Maciejewski, R. Hafen, S. Rudolph, G. Tebbetts, W. Cleveland, S. Grannis, and D. Ebert. Generating synthetic syndromic-surveillance data for evaluating visual-analytics techniques. *Computer Graphics and Applications, IEEE*, 29(3):18–28, May 2009.

[22] R. Maciejewski, R. Hafen, S. Rudolph, G. Tebbetts, W. S. Cleveland, S. J. Grannis, and D. S. Ebert. Generating synthetic syndromic-surveillance data for evaluating visual-analytics techniques. *IEEE Computer Graphics and Applications*, 29(3):18–28, 2009.

[23] R. Maciejewski, S. Rudolph, R. Hafen, A. M. Abusalah, M. Yakout, M. Ouzzani, W. S. Cleveland, S. J. Grannis, and D. S. Ebert. A visual analytics approach to understanding spatiotemporal hotspots. *IEEE Transactions on Visualization and Computer Graphics*, 16(2):205–220, 2010.

[24] R. Maciejewski, S. Rudolph, R. Hafen, A. M. Abusalah, M. Yakout, M. Ouzzani, W. S. Cleveland, S. J. Grannis, M. Wade, and D. S. Ebert. Understanding syndromic hotspots - a visual analytics approach. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technologoy*, pages 35–42, 2008.

[25] A. Malik, R. Maciejewski, T. F. Collins, and D. S. Ebert. Visual analytics law enforcement toolkit. In *Proceedings of the IEEE Conference on Technologies for Homeland Security*, pages 222–228, 2010.

[26] A. Malik, R. Maciejewski, B. Maule, and D. S. Ebert. A visual analytics process for maritime resource allocation and risk assessment. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 221–230, 2011.

[27] K. Matkovic, D. Gracanin, M. Jelovic, A. Ammer, A. Lez, and H. Hauser. Interactive visual analysis of multiple simulation runs using the simulation model view: Understanding and tuning of an electronic unit injector. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1449–1457, 2010.

[28] K. Matkovic, D. Gracanin, M. Jelovic, and Y. Cao. Adaptive interactive multi-resolution computational steering for complex engineering systems. In *Proceedings of the EuroVis Workshop on Visual Analytics*, pages 45–48, 2011.

[29] J. D. Mulder, J. J. van Wijk, and R. van Liere. A survey of computational steering environments. *Future Generation Computer Systems*, 15(1):119–129, 1999.

[30] C. Ncube. On the engineering of systems of systems: key challenges for the requirements engineering community. In *Proceedings of the IEEE Workshop on Requirements Engineering for Systems, Services and Systems-of-Systems*, pages 70–73, 2011.

[31] R. Perez. Supply chain model, April 2011.

[32] Z. Pousman, J. T. Stasko, and M. Mateas. Casual information visualization: Depictions of data in everyday life. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1145–1152, 2007.

[33] H. Ribicic, J. Waser, R. Fuchs, G. Bloschl, and E. Gröller. Visual analysis and steering of flooding simulations. *IEEE Transactions on Visualization and Computer Graphics*, 19(6):1062–1075, 2013.

[34] G. Satell. Why our numbers are always wrong. *Digital Tonto*, October 2012.

[35] G. Satell. Why the future of innovation is simulation. *Forbes*, July 2013.

[36] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, 1986.

[37] S. Terzi and S. Cavalieri. Simulation in the supply chain context: a survey. *Computers in industry*, 53(1):3–16, 2004.

[38] J. J. Thomas and K. A. Cook. *Illuminating the Path: The R&D Agenda for Visual Analytics*. IEEE Press, 2005.

[39] C. Tominski, J. Abello, and H. Schumann. Axes-based visualizations with radial layouts. In *Proceedings of the ACM Symposium on Applied Computing*, pages 1242–1247, 2004.

[40] J. S. Vetter and K. Schwan. Progress: A toolkit for interactive program steering. Technical Report GIT-CC-95-16, Georgia Institute of Technology, 1995.

[41] J. Waser, R. Fuchs, H. Ribicic, B. Schindler, G. Bloschl, and E. Gröller. World lines. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1458–1467, 2010.

[42] J. Waser, H. Ribicic, R. Fuchs, C. Hirsch, B. Schindler, G. Bloschl, and E. Gröller. Nodes on ropes: A comprehensive data and control flow for steering ensemble simulations. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):1872–1881, 2011.