

COLOR CORRECTION FOR OBJECT TRACKING ACROSS MULTIPLE CAMERAS

Satyam Srivastava, Ka Ki Ng, Edward J. Delp

Video and Image Processing Laboratory (VIPER)
School of Electrical and Computer Engineering
Purdue University
West Lafayette, Indiana, USA

ABSTRACT

Color is a powerful attribute that is used to characterize objects for tracking and other surveillance tasks. Since color is dependent on ambient illumination and the imaging equipment, the reliability of color features decreases as the object moves through regions observed with different cameras and with different illumination conditions. Typically, this issue is addressed using alternative color spaces or highly simplified color transformation. In this paper, we propose a more methodical approach to achieving color consistency using colorimetric principles. We model the image analysis system as an observer and develop camera-specific transformations so that images of the same object appear similar to this observer. The transformations thus developed are approximated with 3D look-up tables for fast operation. Experimental results show improved color consistency even under two very different illumination conditions. The results are evaluated both qualitatively and in terms of the effect on a particle filter based object tracker.

Index Terms— surveillance, multi-camera, color consistency, colorimetry

1. INTRODUCTION

Video surveillance is a popular tool used by law enforcement and security personnel. The deployment of multiple cameras is particularly useful for remotely observing extended areas. These cameras may have different characteristics and may be operating under dissimilar illumination conditions. Thus, their outputs will exhibit considerable difference even when imaging the same object. While the white balancing algorithm in many cameras tries to adjust the colors, its function may be affected by the foreground object. This is an undesirable situation for video analysis purposes.

Many image analysis tasks (such as tracking and identification) utilize color features because they are more invariant to shape and orientation changes. However, the variability across the outputs of many cameras makes color a less reliable property when observing a subject over an extended period of time and/or region in space. For example, a parking facility may be interested in tracking a violating vehicle even after it exits the parking area, but this task may be difficult if the lighting conditions inside the parking area are significantly different from the outside.

Some approaches construct more robust color features (histograms and correlograms) in different color space, such as HSV

and LUV. Brown obtained improved accuracies with RGB correlograms in a color object retrieval application [1]. Porikli proposed an inter-camera calibration technique based on a correlation matrix analysis [2]. Gilbert and Bowden describe an incremental learning approach to determining the inter-camera transformation [3] for every camera pair in a network. A conversion vector based method was described by Choi *et al.* and shown to produce promising results [4]. This approach is very suitable for real-time video applications because of its simplicity.

In this paper, we propose a systematic approach to solving the problem of color consistency using principles of colorimetry and color management for imaging and display devices [5]. We identify the effects of camera and illumination conditions in Section 2. We then propose a model-based approach for matching a test device's output to that of a reference device in Section 3. In the same section, we devise a faster look-up table based realization of the model-based transformation. These methods are tested by color correcting several objects under two imaging conditions and compared with the conversion vector method [4]. These are provided in Section 4. We summarize the work and propose some future extensions in Section 5.

2. COLOR CORRECTION

Consider a public site (such as an airport or a shopping mall) which is surveilled with a network of k video cameras, possibly under k different illumination conditions. In this work, we assume that each illumination remains fairly constant over time. Thus, we can combine a camera-illumination pair into a single entity which we denote as an *imaging unit* (IU). Let the output of the k IUs at any time t be represented by $I_0(t), I_1(t), \dots, I_{k-1}(t)$. Note that each $I_r(t)$ consists of pixel-wise color co-ordinates for the scene. For all practical purposes, we can safely assume that the color at a pixel is specified as an RGB value in the camera output. Further, we note that it is unrealistic to assume that an object would be imaged at the same time by all the IUs. Therefore, we drop the time index in subsequent discussions.

It is natural to expect significant variation in the images (I_r) of an object generated by the different imaging units. If the subject entered the surveillance system from IU a and subsequently moved into the field of IU b , one would like to transform I_b such that the colors resemble those in I_a . Thus, we would need an RGB-RGB transform for every pair of imaging units in the system.

Our first step to solving this problem is to prevent the quadratic growth in the number of transforms needed ($\binom{k}{2}$ for k IUs). We do this by modeling the image analysis system as an "observer." Note that while the term observer is inspired by colorimetry, its interpretation here is very different from the CIE standard observer [6]. Our

This paper is based upon work supported by the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0001. Address all correspondence to E. J. Delp (ace@ecn.purdue.edu).

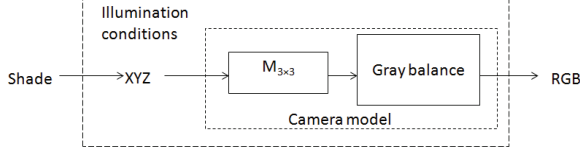


Fig. 1. A schematic diagram of an imaging unit.

observer can be viewed as a standard camera operating under chosen illumination conditions – in other words, an imaging unit as defined above. Therefore, our goal is to match the output of every IU with that of the reference IU. The problem now scales linearly with the number of IUs. Without loss of generality we designate I_0 as the output from the reference IU.

Since the effects of illumination are abstracted into the IUs, the only invariant property of a subject would be its surface reflectance. However, to make the discussion more intuitive, we refer to this as “shade.” Shade is the identifier that a human would associate with an object with such surface reflectance and viewed under typical illumination. Thus, we require that subjects with the same shade produce similar RGB values in all IUs. The actual relation between the shade and the observed tri-stimuli (under given illumination) is difficult to establish and requires extensive radiometric measurements. Instead, we make this key assumption that the two quantities are related through the illumination white point. The resultant tri-stimuli in CIE XYZ would be the input to the camera which itself can be modeled with gray balancing and a linear transformation [7]. The complete model of an IU is illustrated in Figure 1.

2.1. Tracking with color features

Particle filter is popularly used for object tracking because it has been shown to be very successful for non-linear and non-Gaussian dynamic state estimation problems and is very reliable in cases like clutter and occlusions [8]. Each hypothesized state is referred to as a particle and a weighted sum of all particles gives the final estimate of the state. An observation likelihood model assigns each particle a weight according to how this particle resembles the target object. This model or rule is determined quantitatively by measuring the dissimilarity between the target q and the particle p . We use the Bhattacharyya distance as the metric of distance which is given by:

$$d[p, q] = \sqrt{1 - \sum_{u=1}^m \sqrt{p_u q_u}}, \quad (1)$$

where u is the bin index of the color distributions. The particle weights are approximated using a Gaussian distribution of the Bhattacharyya distances. Similar distributions result in smaller $d[p, q]$.

Color distributions have been widely used for tracking problems [9] because they are robust to partial occlusion, and are rotation and scale invariant. The color distribution is expressed by an m -bin histogram, whose components are normalized so that the sum of all bins equals one. For a region A in an image, given a set of n pixels in A denoted by $B = \{x_i, i = 1, 2, \dots, n\} \in A$, the m -bin color histogram $T(A) = \{h_j, j = 1, 2, \dots, m\}$ can be obtained by assigning each pixel, x_i to a bin, by the following equation:

$$h_j = \frac{1}{n} \sum_{x_i \in B} \delta_j[b(x_i)], \quad (2)$$

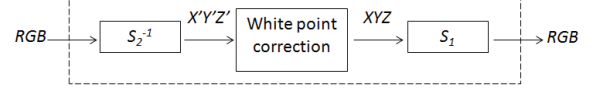


Fig. 2. A block diagram of the device to device transformation CCMX (Φ).

where $b(x_i)$ is the bin index in which the color component at x_i falls and δ is the Kronecker delta function.

3. DETAILED FORMULATION

An advantage of selecting a reference is that the discussion and modeling can be build for two camera systems ($k = 2$) and extended to multi-camera systems without any changes. In this section, we first describe a model-based method to match the output of a test imaging unit (I_1) with that of a reference IU (I_0), and then approximate this transformation using 3D look-up tables (LUT). This two-step approach to solving a color correction problem was also proposed by Srivastava *et al.* for color management of display devices [10].

3.1. Model-Based Transformation

Let \mathfrak{S} represent the commonly used RGB color space. This is distinct from the linear RGB representation, where the values are proportional to the photon count. The latter are explicitly marked with a superscript (for example, R_n^{lin}). We require a function $\Phi : \mathfrak{S} \rightarrow \mathfrak{S}$ such that $\Phi(I_1) = I_0$, for the same shade. If the camera model is represented by S , then Figure 2 illustrates the construction of Φ . This method is denoted by CCMX (Camera to Camera Model-Based Transformation) in this paper. The white correction step consists of a technique to account for the different illumination conditions.

We choose a camera model which consists of gray balancing and a linear transform. Thus, a camera model S transforms a stimulus $(X, Y, Z)^T$ to $(R, G, B)^T$ as follows:

$$\begin{bmatrix} R_n^{lin} \\ G_n^{lin} \\ B_n^{lin} \end{bmatrix} = \begin{bmatrix} & & \\ & M_{3 \times 3} & \\ & & \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}. \quad (3)$$

Here $(R, G, B)_n^{lin}$ signify the normalized linear values and M is a linear transform. Absolute linear values can be obtained by scaling with the absolute linear values of the device white. This can also be represented using the inverse but more intuitive normalization operation,

$$R_n^{lin} = \frac{R_{abs}^{lin}}{R_{abs}^{lin w}}; G_n^{lin} = \frac{G_{abs}^{lin}}{G_{abs}^{lin w}}; B_n^{lin} = \frac{B_{abs}^{lin}}{B_{abs}^{lin w}}, \quad (4)$$

where $(R, G, B)_{abs}^{lin}$ represent the absolute linear values which are proportional to the photon count at the camera sensor. Finally, the device RGB output is obtained by gamma correction or gray balancing [11]. This can be represented by a 3D function

$$f : \mathfrak{R}^3 \rightarrow \{0, 1, \dots, 255\}^3. \quad (5)$$

In our modeling, we use a gain-gamma-offset model [12] for the function f , and obtain the transformation matrix M by linear regression. The parameters of the models are computed by measuring the RGB and CIE XYZ values for a small number of printed patches.



Fig. 3. A printed sheet of colored patches used to construct the camera models.

We used only 24 colored patches (shown in Figure 3) and measured the XYZ values with a spectroradiometer PR-705.

While many white point compensation techniques have been proposed in the literature [13], we use a simple rescaling based approach because it is easy to apply in real situations with very little information about the illumination available. Let the reference illumination white point be $(X, Y, Z)_w^0$ and the test white point be $(X, Y, Z)_w^1$. Then to transform an object's tristimulus (X', Y', Z') to the reference conditions, we obtain

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \frac{X_w^0}{X_w^1} & 0 & 0 \\ 0 & \frac{Y_w^0}{Y_w^1} & 0 \\ 0 & 0 & \frac{Z_w^0}{Z_w^1} \end{bmatrix} \begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix}. \quad (6)$$

3.2. 3D Look-Up Table

Our goal is to efficiently color correct the output I_1 . It has been shown that look-up tables (LUT) can be used to achieve complex transformations [14]. LUTs are suitable for hardware implementations and allow elaborate modeling of the system they approximate. This is because we can improve the function Φ at the cost of added complexity. But the use of LUTs makes this additional complexity inconsequential.

Let the table look-up operation be represented by a function $\mathcal{L} : \mathfrak{S} \rightarrow \mathfrak{S}$. If the function Φ has a domain D and range S , then we evaluate Φ at certain points given by $D_1 \subset D$. In our application, we require a 3D LUT whose output is also a 3-tuple. Consider a LUT of size $n_R \times n_G \times n_B$, then

$$D_1 = \{r_0, r_1, \dots, r_{n_R-1}\} \times \{g_0, g_1, \dots, g_{n_G-1}\} \quad (7)$$

$$\times \{b_0, b_1, \dots, b_{n_B-1}\}, \quad (8)$$

where \times represents a Cartesian product. The output of the LUT is given by

$$\mathcal{L}([r, g, b]) = \begin{cases} \Phi([r, g, b]), & \forall [r, g, b] \in D_1, \\ H([r, g, b]), & \text{otherwise.} \end{cases} \quad (9)$$

Here H represents interpolation which is used to estimate the value of Φ using the known values at the neighboring points in D_1 .

4. EXPERIMENTAL RESULTS

The methods developed above were tested for a system with two imaging units. We used a single video camera (Sony DCR-TRV33) to image objects under two different illumination conditions. The controlled illumination was achieved using a Gretag MacBeth light booth and the two illuminants (Daylight and Horizon) were selected to approximately resemble the outside and inside lighting of a typical parking garage. Note that we directly use the native RGB output of the cameras and do not require availability of special formats (such as RAW).

Table 1. Mean errors (Δ) between the reference image and the corrected images.

Test \ Method	None	Offset	LUT	CCMX
Golf	22.86	23.65	16.58	10.46
DVD	27.57	27.72	19.47	15.54
Phone	37.05	20.01	18.02	10.17
Patch	15.00	25.83	9.67	5.01

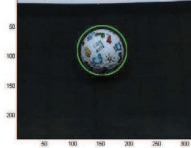


Fig. 4. An illustration of the target region used to compute the color histogram.

Since our objective is to make the corrected colors $(\tilde{R}, \tilde{g}, \tilde{B})$ closer to the reference colors (R, G, B) in the RGB space, we use a Euclidean metric of distance given by:

$$\Delta = \|(\tilde{R}, \tilde{g}, \tilde{B}) - (R, G, B)\|. \quad (10)$$

Table 1 shows the mean error (Δ) values for different objects and corrected using the proposed techniques. We also corrected the colors using a constant offset in every channel as proposed by Choi [4].

In order to test the effect on color tracking, we estimate the improvement in the observation likelihood model due to color correction. Recall that the suitability of a particle is measured using the Bhattacharyya distance. Thus, we compute the Bhattacharyya distance of the object under the reference and the test conditions with different correction methods. The color features are extracted only in a small elliptical region around the object as shown in Figure 4. The distances for different test objects are presented in Table 2. As stated in Section 2.1 a lower distance would result in the particle being assigned a higher weight. In these experiments, the particle actually contains the target. Therefore, a higher weight (and lower distance) implies improved performance of the tracker.

There are three important observations in these results. It is evident that the color features become more robust after correction with our proposed methods. Secondly, the offset method [4] can sometimes result in more error than the uncorrected image. Finally, none

Table 2. Bhattacharyya distances of the candidate object in the reference and corrected images.

Test \ Method	None	Offset	LUT	CCMX
Golf	0.81	0.67	0.35	0.19
DVD	0.56	0.82	0.29	0.22
Phone	0.58	0.86	0.39	0.32
Patch	1.00	1.00	1.00	1.00



Fig. 5. Qualitative results using different color correction techniques. The columns represent (from left to right) (i) the test image, (ii) output of the offset method, (iii) output of the 3D LUT method, (iv) output of the CCMX method, and (v) the reference image.

of the methods appear to work when correcting a constant color patch with our chosen feature ($8 \times 8 \times 8$ histogram). This is not entirely unexpected because of the high granularity of our histogram bins. The corrected color may fall into a different bin even when its Euclidean distance from the reference is only 5 units. Since the histogram in this case will be an impulse function, the Bhattacharyya coefficient would be zero. Such degeneracy can be avoided by “soft” assignment to histogram bins but this has not been used in this paper.

The resultant images for two of the test cases are shown in Figure 5. Due to the nature of the results, these figures are better viewed in color. From these results we can conclude that the proposed methods offer improved color correction in both quantitative and qualitative tests. The LUT-based method slightly increases the error as compared to the CCMX method but is considerably more accurate than the offset method.

5. CONCLUSIONS

In this paper, we described the problem of inconsistent color features when a subject is tracked (or otherwise observed) through different cameras and illumination conditions. We isolated the effects of illumination and camera characteristics so as to establish a mapping between the object’s “shade” and the camera’s RGB output. We used the basic camera models to devise a method for matching a given camera’s output under certain level of illumination to a reference camera’s output under reference illumination. We proposed realizing this method using a fast table look-up operation. Our proposed methods were compared with a popular color correction technique and shown to produce better results. This work can be extended by using more elaborate white point compensation techniques and non-linear camera models. Also, this method can be tested by tracking objects in real multi-camera systems.

6. REFERENCES

- [1] L. M. Brown, “Example-based color vehicle retrieval for surveillance,” *Proceedings of IEEE International Conference on Advanced Video and Signal-Based Surveillance*, Boston, Massachusetts, August-September 2010, pp. 91–96.
- [2] F. Porikli, “Inter-camera color calibration by correlation model function,” *Proceedings of IEEE International Conference on Image Processing*, Barcelona, Spain, September 2003, pp. 133–136.
- [3] A. Gilbert and R. Bowden, “Tracking objects across cameras by incrementally learning inter-camera colour calibration and patterns of activity,” *Proceedings of European Conference on Computer Vision*, Graz, Austria, May 2006, pp. 125–136.
- [4] Y. Choi, Y. Lee, and W. Cho, “Color correction for object identification from images with different color illumination,” *Proceedings of International Conference on Networked Computing and Advanced Information Management*, Seoul, Korea, August 2009, pp. 1598–1603.
- [5] B. Fraser, C. Murphy, and F. Bunting, *Real World Color Management*. Berkeley, California: Peachpit Press, 2004.
- [6] CIE, *Commission Internationale de l’Eclairage Proceedings*. Cambridge, Massachusetts: Cambridge University Press, 1931.
- [7] F. M. Verdu, J. Pujol, and P. Capilla, “Characterization of a digital camera as an absolute tristimulus colorimeter,” *Journal of Imaging Science and Technology*, vol. 47, no. 4, pp. 279–295, July-August 2003.
- [8] K. K. Ng and E. J. Delp, “New models for real-time tracking using particle filtering,” *Proceedings of SPIEIS&T Electronic Imaging: Visual Communications and Image Processing*, vol. 7257, San Jose, California, January 2009, pp. 72 570B:1–12.
- [9] K. Nummiaro, E. Koller-Meier, and L. V. Gool, “Color features for tracking non-rigid objects,” *Chinese Journal of Automation - Special Issue on Visual Surveillance*, vol. 29, no. 3, pp. 345–355, May 2003.
- [10] S. Srivastava, T. H. Ha, J. P. Allebach, and E. J. Delp, “Color management using optimal three dimensional look-up tables,” *Journal of Imaging Science and Technology*, vol. 54, no. 3, pp. 402:1 – 402:14, May-June 2010.
- [11] C. Poynton, *Digital Video and HDTV: Algorithms and Interfaces*. San Francisco, California: Morgan Kaufmann, 2003.
- [12] C. Yang-Ho, I. M. Hye-Bong, and H. A. Yeong-Ho, “Inverse characterization method of alternate gain-offset-gamma model for accurate color reproduction in display devices,” *Journal of Imaging Science and Technology*, vol. 50, no. 2, pp. 139–148, March-April 2006.
- [13] G. Sharma, Ed., *Digital Color Imaging Handbook*. Boca Raton, Florida: CRC Press, 2002.
- [14] S. Srivastava, T. H. Ha, E. J. Delp, and J. P. Allebach, “Generating optimal look-up tables to achieve complex color space transformations,” *Proceedings of the IEEE International Conference on Image Processing*, Cairo, Egypt, November 2009, pp. 1641–1644.