

A robust incremental learning framework for accurate skin region segmentation in color images[☆]

Bin Li^{a,*}, Xiangyang Xue^a, Jianping Fan^b

^aDepartment of Computer Science and Engineering, Fudan University, Shanghai 200433, China

^bDepartment of Computer Science, UNC-Charlotte, Charlotte, NC 28223, USA

Received 5 July 2006; received in revised form 3 April 2007; accepted 29 April 2007

Abstract

In this paper, we propose a robust incremental learning framework for accurate skin region segmentation in real-life images. The proposed framework is able to automatically learn the skin color information from each test image in real-time and generate the specific skin model (SSM) for that image. Consequently, the SSM can adapt to a certain image, in which the skin colors may vary from one region to another due to illumination conditions and inherent skin colors. The proposed framework consists of multiple iterations to learn the SSM, and each iteration comprises two major steps: (1) collecting new skin samples by region growing; (2) updating the skin model incrementally with the available skin samples. After the skin model converges (i.e., becomes the SSM), a post-processing can be further performed to fill up the interstices on the skin map. We performed a set of experiments on a large-scale real-life image database and our method observably outperformed the well-known Bayesian histogram. The experimental results confirm that the SSM is more robust than static skin models.

© 2007 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.

Keywords: Skin region segmentation; Specific skin model; Incremental learning; Generic to specific

1. Introduction

Color-based skin detection and segmentation techniques have broad applications and these applications fall into two categories: (1) fast skin pixels detection (e.g., face localization in images and face tracking in video sequences). These applications are not rigorous on the integrities of the detected regions, but they focus on fast locating the potential skin blobs. (2) Accurate skin region segmentation (e.g., gesture recognition and objectionable image filtering). These applications require the segmentation results to retain more semantic information, such as shape and structure, which are always used as the primitives for the subsequent feature extraction.

In the last decade, a large number of skin detection techniques have been reported. Vezhnevets et al. [1] made a

comprehensive survey on the pixel-based skin detection techniques and Phung et al. [2] studied various state-of-the-art skin modeling and classification algorithms. The reviewed methods in the two surveys involve classifying individual pixels into skin and non-skin categories by a static statistical model (e.g., Gaussian mixture model (GMM)) or a predefined classifier (e.g., multi-layer perceptron), which is learned offline from a large collection of skin pixels. Most of these methods can be used to fast detect potential skin blobs and they satisfy the first application category. However, the static skin detectors cannot automatically adapt to a certain test image in which the skin colors may vary from one region to another due to illumination conditions and inherent skin colors; thus they are unsuitable for accurate skin region segmentation.

The diversity of skin colors in a certain image is mainly induced by inherent skin colors and illumination conditions. For inherent skin colors, the skins of different persons or different parts on a human body may get different colors. For illumination conditions, skin colors are roughly influenced in three ways: the skin surface may (1) lighten in highlight areas and deepen in shadow areas under directional lights, (2) reflect the

[☆] This work was supported in part by the National Natural Science Foundation of China (60533100, 60402007) and Shanghai Municipal R&D Foundations (05QMH1403, 065115017, and 06DZ15008).

* Corresponding author. Tel.: +86 21 6564 3720; fax: +86 21 6564 2820.

E-mail addresses: libin@fudan.edu.cn (B. Li), xyxue@fudan.edu.cn (X. Xue), jfan@uncc.edu (J. Fan).

colors of the objects nearby, and (3) reflect the global and uniform ambient light. Human vision system is insensitive to these skin color variations due to the “color constancy” phenomenon [3], but digital equipments can capture these variations objectively. Unfortunately, such “objectivity” makes static skin detectors helpless to segment an accurate skin map since some skin regions in a certain image are of distorted colors caused by the above factors.

Due to the uncertainties of skin color variations, the techniques for accurate skin region segmentation are not as many as the ones for fast skin pixels detection. Martinkauppi et al. [4] compared some adaptive methods [5–7] for skin detection under changing illuminations. Hsu et al. [6] first performed lighting compensation, then used a statistical skin model to detect the candidate skin regions, and finally located the faces based on facial features. Soriano et al. [7] initiated a “skin locus” histogram with the skin pixels extracted from the first frame of a video clip; in the tracking process, the histogram was updated with the skin pixels detected in the current frame to adapt to the illumination conditions in the following frames. Some earlier works [8,9] have the similar frameworks like Ref. [7] for face tracking. Another two methods [10,11] were also developed to construct adaptive skin models. In Ref. [10], an image is first segmented into homogeneous regions, and the pixels in each region are examined by a neural network; then the pixels in the regions which get higher skin coverage ratios are used to train a skin model. In Ref. [11], the potential skin pixels in an image are first detected to train a GMM; based on the assumption that there must be one compact Gaussian in the obtained GMM being able to fit the real skin distribution in the image, a predefined SVM is used to identify the true Gaussian based on the shapes of the detected regions.

Although the adaptive methods reviewed above are able to work well in certain scenarios, they can hardly be applied as a general framework for accurate skin region segmentation in color images. Most of these methods have limitations: Refs. [7–9] can only work for video clips, Refs. [6,11] require domain knowledge (e.g., facial features), and Ref. [10] largely depends on an accurate image segmentation result, but image segmentation itself is also a difficult problem.

In this paper, we propose an unsupervised incremental learning framework for accurate skin region segmentation in real-life color images (e.g., press photography). The proposed method is able to automatically learn a specific skin model (SSM) for each test image in real-time such that the SSM can adapt to the varying skin colors in that image. Different from most of the reported adaptive methods, our method can be used to detect not only faces but also any exposed regions on the human bodies in both color images and videos, thanks to the only simple assumption we adopt; i.e., a small patch of skin pixels (seeded region) can be correctly detected in an image by a generic skin model.

The remainder of the paper is organized as follows: In Section 2, we first state the motivation of our work. Then we present the incremental learning framework and describe the required techniques in Section 3. In Section 4, an incremental learning algorithm for statistical model is introduced. A post-processing

named boundary potential field is also introduced in Section 5. The experimental results are demonstrated in Section 6, and finally we conclude this paper in Section 7.

2. Motivation

A generic statistical skin model learned from a large training set is able to obtain an optimal average performance over large amounts of images, but it cannot provide the best performance for each image. This situation can be illustrated in Fig. 1, where a generic statistical model (GMM) is used to detect skin pixels. If higher true positive rate (i.e., recall) is stressed (see Fig. 1(c)), higher false positive rate (i.e., false alarm) is induced; if lower false positive rate is stressed (see Fig. 1(d)), more false negative (i.e., missing examples) are induced. It is impossible for the generic skin model to simultaneously improve the true positive rate and reduce the false positive rate. There is a tradeoff between these two rates since the generic skin model focuses on the statistical property of skin colors in large amounts of images other than a single image. However, if each test image has one SSM learned from itself, the performance on the corresponding image can be optimized. Accordingly, the average performance over the whole test image set also can be optimized.

Based on the above analysis, we find that the performance on each test image can be optimized by addressing the following two problems:

- *How to learn the SSM for an image?:* We consider a “generic to specific” skin model evolution. For an image containing skin regions, we assume that several (at least one) small patches of skin pixels can be detected by a generic skin model¹ (a statistical skin model learned offline). By treating these detected skin patches as the seeded regions, a run of seeded region growing (SRG) can be performed to collect more skin samples from that image. As a result, new skin samples can be used to update the generic skin model to approximate the real skin distribution in that image.
- *How to adapt to the skin color variations in an image?:* SRG is able to collect adjoining skin patches with distorted colors, but one run of region growing is insufficient. A common case in practice is illustrated in Fig. 2. An image contains three disconnected skin regions with different color distributions. A2 and B2, B3 and C3 have similar color distributions. Suppose the colors of skin pixels in A1 are regular and A1 is detected by a generic skin model, thus the new skin pixels in A2 can be obtained by region growing. Then the generic skin model is updated and the updated skin model is able to detect A1, A2, and B2. Another run of region growing originated from A1, A2, and B2 obtains the new skin pixels in B3, and the skin model is updated again. Subsequently, the updated skin model is able to detect all the skin regions and it becomes the SSM for that image.

¹ The generic skin model may fail in the case that the entire image is in a distorted color due to, for example, incorrect white balance.

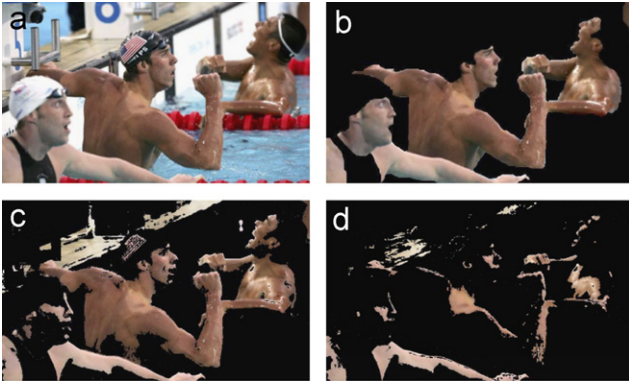


Fig. 1. The limitation of the generic skin model. (a) Original image, (b) ground truth, (c) loose threshold, and (d) tight threshold.

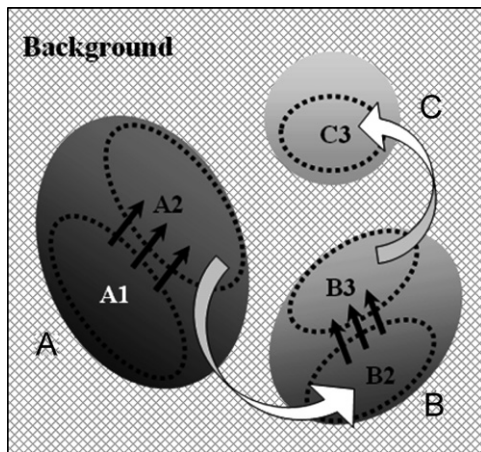


Fig. 2. The motivation of incremental learning iterations. A, B, and C denote three disconnected skin regions; 1, 2, and 3 denote three different color distributions.

3. Framework for SSM generation

In this section, we introduce the proposed framework for accurate skin region segmentation. The inputs of the framework are a test image obtained online and a generic statistical skin model learned offline; the outputs are the segmentation result (skin map) and the SSM (which has been used to generate the skin map). The proposed incremental learning framework follows a “generic to specific” evolution to estimate and update the skin model over time.

The entire process starts with a pixel-wise skin detection by a generic skin model on an image like most skin detection methods do. The difference is that a relative lower false positive rate is stressed (e.g., using a tighter threshold) in our method instead of a tradeoff between the true and the false positive rates. We aim at screening out confident skin patches as the primary seeded regions (PSRs) for skin region growing. After obtaining the PSRs, multiple incremental learning iterations start to update the skin model incrementally. Each iteration comprises two major steps:

- *Sample-collection step*: The skin pixels which are detected by the updated skin model (by the generic skin model at the

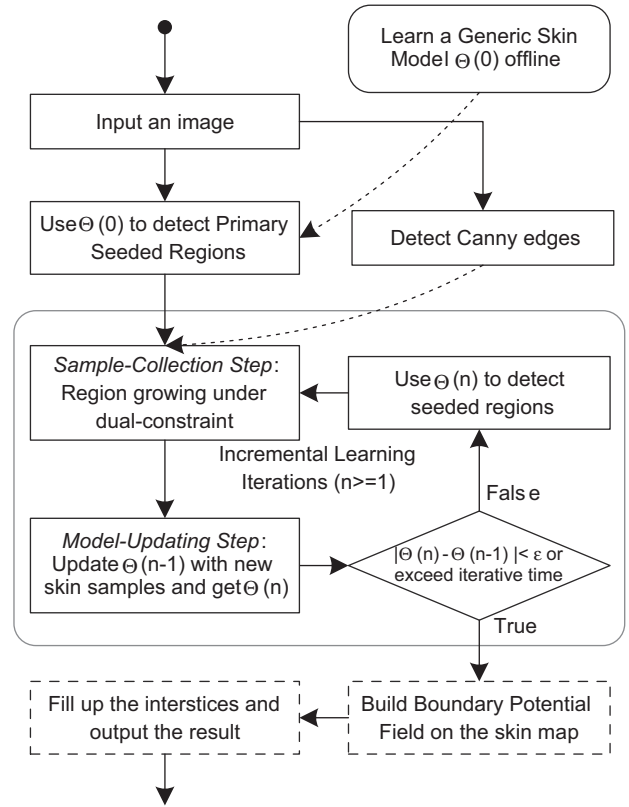


Fig. 3. Incremental learning framework.

first iteration) are treated as the seeded regions, and a run of region growing controlled by both edge and color information is performed to collect the new skin pixels (illustrated by the parallel arrows in Fig. 2). The obtained skin pixels will be used to update the underlying skin model in the model-updating step.

- *Model-updating step*: The skin model is updated with the new skin pixels obtained in the sample-collection step and it becomes more accurate to approximate the real skin distribution in the image. Then, the updated skin model is used to detect the skin regions again, hence more integrated skin regions can be obtained and, moreover, the disconnected skin regions (illustrated by the hollow arrows in Fig. 2) can be also detected.

Phung et al. [2] and Albiol et al. [12] have proved that all the color spaces have equal capabilities to represent the skin colors. For the applications to videos (e.g., MPEG), we choose YUV color space in this paper. The skin models we adopt are GMMs which are denoted by $\Theta(\cdot)$. We let $\Theta(0)$ denote the generic skin model and $\Theta(n)$ the skin model at the n th incremental learning iteration. Fig. 3 is the flowchart of the incremental learning framework which consists of four phases (the dashed rectangles can be omitted for a more general and simple framework):

- (1) *Offline-process*: The EM algorithm [13] is used to learn the generic skin model $\Theta(0)$ offline on a large training set of skin pixels.



Fig. 4. Incremental learning iterations. Seeded regions are denoted by white patches, and region growing results are denoted by shaded patches. (a) Original image, (b) initial skin map, (c) primary seeded regions, (d) region growing (first iteration), (e) skin map (first iteration), (f) seeded regions (first iteration), (g) region growing (second iteration), and (h) skin map (second iteration).

- (2) *Pre-process*: For an input image, Canny edge [14] detection is performed to detect the edges in the image, and then the generic skin model $\Theta(0)$ is used to detect the PSRs in the image.
- (3) *Incremental-process*: When the PSRs are obtained, they are treated as the seeds for region growing. The new skin pixels are then used to update $\Theta(0)$ and the updated skin model becomes $\Theta(1)$. The sample-collection step and model-updating step are repeated until no more new skin pixels in the image can be collected (i.e., the skin model converges: $|\Theta(n) - \Theta(n-1)| < \varepsilon$) or exceed the iterative time, then the SSM for the image is obtained.
- (4) *Post-process*: The boundary potential field is built on the skin map generated by the SSM at the final iteration, and it guides the region growing flow to fill up the interstices.

3.1. Generic skin model

The EM algorithm [13] is employed to estimate the parameters of the generic skin model. $\Theta = \{(w_m, \mu_m, \Sigma_m)\}_{m=1}^M$ is a GMM, where M denotes the number of Gaussians in the mixture, w_m , μ_m , and Σ_m , respectively, denote the weight, the mean vector, and the covariance matrix of the m th Gaussian. We use Bayesian information criterion (BIC) [15] to determine the number of Gaussians in the mixture. Fig. 5(a) plots the distribution of the generic skin model.

3.2. Edge detection

Canny operator [14] has two thresholds, $T1$ and $T2$ ($T1 > T2$). Tracking can only begin at a point on the ridge higher than $T1$ and continues until the point falls below $T2$. We set $T1 = 0.9$ and $T2 = 0.1$ in order to detect the salient edges and keep their integrities, and discard the trivial edges.

3.3. Region growing algorithm

In the sample-collection step, skin region growing is performed under two types of constraints, i.e., color and edge.

The “edge” type means that the growing regions should stop when they encounter edges, thus the skin regions would be less likely to grow into the background. We illustrate the region growing process via an example in Fig. 4. At the first incremental learning iteration, the PSRs are screened out in the following way. A sliding window (the size of the sliding window is discussed in Section 6.1) moves on the initial skin map detected by $\Theta(0)$ (see Fig. 4(b)); if it covers a patch whose ratio of skin area reaches 100%, the patch is selected as a PSR. Fig. 4(c) illustrates all the extracted PSRs. At the succeeding iterations, the seeded regions are selected in the same way.

We adapt the algorithm of SRG [16] to implement our skin region growing algorithm, which is described in Algorithm 1, where $\|\cdot\|$ denotes the Euclidean norm and δ the color difference threshold ($\delta = 24$, and the extended YUV range is $[0, 255]$). Fig. 4(d) is the region growing result at the first iteration, where the generic skin model $\Theta(0)$ is updated and $\Theta(1)$ is obtained. Using $\Theta(1)$ to detect the skin regions, one can see that the result (see Fig. 4(e)) is much better than the initial skin map. The skin regions of the two farside swimmers are almost detected and the noises in the background are eliminated. Fig. 4(h) is the even better result at the second iteration.

Algorithm 1. Skin region growing algorithm.

- 1: Push all the neighboring pixels of the seeded regions into the sequentially sorted list (SSL);
- 2: **while** SSL is not empty **do**
- 3: Remove the first pixel \mathbf{z} from the SSL;
- 4: Cover a window (W) located at \mathbf{z} ;
- 5: Calculate $\bar{\mathbf{x}}$ (mean YUV vector) of the labeled pixels in W ;
- 6: Calculate \mathbf{x} (YUV vector) of \mathbf{z} ;
- 7: **if** $\|\bar{\mathbf{x}} - \mathbf{x}\| < \delta$ and no edge crosses W **then**
- 8: Label \mathbf{z} as a skin pixel;
- 9: Push the unlabeled neighboring pixels of \mathbf{z} into the SSL;
- 10: **end if**
- 11: **end while**.

4. Incremental statistical model

In Section 3, one can find that it is the two important steps at each incremental learning iteration that realize the skin model evolution. The remaining problem is, in the Model-updating step, how to update the skin model with the new skin pixels obtained in the sample-collection step. At each iteration, we only have the skin model (the generative model) generated at the last iteration and the new skin pixels obtained at the current iteration. Thus, we should construct the new skin model based on the combination of the original skin model and the newly obtained sample set.

Suppose we get the skin model Θ generated at the last iteration and the new skin sample set $\mathcal{X} = \{\mathbf{x}_l\}_{l=1}^L$, where L denotes the number of the skin pixels. To construct the new skin model based on the combination of Θ and \mathcal{X} , the function should be $\mathcal{F}(\Theta, \mathcal{X})$. Obviously, for online incremental learning, recursive scheme is most efficient. Therefore, \mathcal{F} can be resolved to be the recursive form $\Theta^{(N+1)} = f(\Theta^{(N)}, \mathbf{x}_{N+1})$, and it is worth noting that the superscripts (N) and $(N+1)$ denote the number of samples which have been used for training the skin model Θ , while the subscript $N+1$ denotes the $(N+1)$ th sample. We take the first incremental learning iteration, for example. The generic skin model learned from K skin pixels is denoted by $\Theta^{(K)}$, and L new skin pixels are obtained in the sample-collection step. Then $\mathcal{F}(\Theta^{(K)}, \mathcal{X})$ comprises L loops to generate the new skin model:

$$\Theta^{(N+1)} = f(\Theta^{(N)}, \mathbf{x}_{N+1}), \quad N = K, \dots, (K+L-1). \quad (1)$$

The computational complexity of Eq. (1) is linear.

For the incremental learning algorithm, one should be able to adjust the learning rate. Accordingly, we define the factor λ , which can be regarded as the ratio of the new samples to the total samples, i.e., $\lambda = L/(K+L)$. However, both K and L are fixed, thus we cannot adjust them to obtain the expected value of λ . The order of magnitude of the original samples (K) is large, e.g., 10^7 , while the number of the new samples (L) obtained at each iteration can only reach 10^3 – 10^4 , so λ is far beyond the expected value (e.g., 0.1–0.9). Nevertheless, we can think in another way. Although $\Theta^{(K)}$ is learned from K samples, we can also treat it as a generative model which is fitted by \hat{K} samples, where $\hat{K} \ll K$. Therefore, the learning rate factor becomes $\lambda = L/(\hat{K}+L)$, and we can calculate \hat{K} to satisfy the expected value of λ :

$$\hat{K} = \frac{L}{\lambda} - L. \quad (2)$$

Then, the skin model generated at the last iteration should be denoted by $\Theta^{(\hat{K})}$. Accordingly, the function for generating the new skin model based on the combination of the original skin model and the new skin sample set gives

$$\Theta^{(K+L)} = \mathcal{F}(\Theta^{(\hat{K})}, \mathcal{X}). \quad (3)$$

Eq. (3) is the prototype of the incremental learning algorithm for skin model updating, and it is used at each incremental learning iteration to generate the new skin model $\Theta^{(K+L)}$ based

on the skin model $\Theta^{(\hat{K})}$ generated at the last iteration and the skin sample set \mathcal{X} obtained at the current iteration.

4.1. Incremental learning algorithm

GMMs and histograms are the most widely used statistical models, and they both can be employed to solve Eq. (3). Since the generative model is more simple and space saving than the bins, we adopt GMM in this paper.

Some reported algorithms, e.g., Refs. [17,18], can be used for GMM incremental learning, but these complicated methods are not suitable for our application since they also select the optimal number of Gaussians in the mixture. In our application, the number of Gaussians could be kept invariant for simplicity during the entire Incremental-process, because (1) the real skin distribution in an image must be the subset of the generic skin model, so it is needless to add Gaussians; and (2) the weights of those unconcerned Gaussians will vanish during the process of model updating, so it is needless to reduce Gaussians. Therefore, we consider a simplified version of Ref. [19] for updating the skin model.

Given the skin model $\Theta^{(\hat{K})}$ with M Gaussians generated at the last iteration and L new skin pixels obtained at the current iteration, Eq. (3) can be written in the form of Eq. (1):

$$w_m^{(N+1)} = f_w(w_m^{(N)}, \mathbf{x}_{N+1}), \quad N = \hat{K}, \dots, (\hat{K}+L-1), \quad (4)$$

$$\boldsymbol{\mu}_m^{(N+1)} = f_\mu(\boldsymbol{\mu}_m^{(N)}, \mathbf{x}_{N+1}), \quad N = \hat{K}, \dots, (\hat{K}+L-1), \quad (5)$$

$$\boldsymbol{\Sigma}_m^{(N+1)} = f_\Sigma(\boldsymbol{\Sigma}_m^{(N)}, \mathbf{x}_{N+1}), \quad N = \hat{K}, \dots, (\hat{K}+L-1), \quad (6)$$

where w_m , $\boldsymbol{\mu}_m$, and $\boldsymbol{\Sigma}_m$, respectively, denote the weight, the mean vector, and the covariance matrix of the m th Gaussian in $\Theta^{(\hat{K})}$.

The recursive equations for updating GMM are derived from the basic equations for calculating the weights, the mean vectors, and the covariance matrices of GMM. Firstly, the posterior probability (ownership) of \mathbf{x}_n , given \mathbf{x}_n belonging to the m th Gaussian, should be defined:

$$P(m|\mathbf{x}_n, \Theta_m^{(N)}) = \frac{w_m^{(N)} p_m(\mathbf{x}_n|\Theta_m^{(N)})}{\sum_{j=1}^M w_j^{(N)} p_j(\mathbf{x}_n|\Theta_j^{(N)})}, \quad (7)$$

where $p_m(\cdot)$ denotes the probability density function of the m th Gaussian. It is worth noting that the superscript (N) means the number of samples as mentioned before, other than the iterative time in the EM algorithm. Given Eq. (7), the basic equations for calculating the parameters of the m th Gaussian are defined as

$$w_m^{(N)} = \frac{\sum_{n=1}^N P(m|\mathbf{x}_n, \Theta_m^{(N)})}{N}, \quad (8)$$

$$\boldsymbol{\mu}_m^{(N)} = \frac{\sum_{n=1}^N \mathbf{x}_n P(m|\mathbf{x}_n, \Theta_m^{(N)})}{\sum_{n=1}^N P(m|\mathbf{x}_n, \Theta_m^{(N)})}, \quad (9)$$

$$\boldsymbol{\Sigma}_m^{(N)} = \frac{\sum_{n=1}^N (\mathbf{x}_n - \boldsymbol{\mu}_m^{(N)})(\mathbf{x}_n - \boldsymbol{\mu}_m^{(N)})^T P(m|\mathbf{x}_n, \Theta_m^{(N)})}{\sum_{n=1}^N P(m|\mathbf{x}_n, \Theta_m^{(N)})}. \quad (10)$$

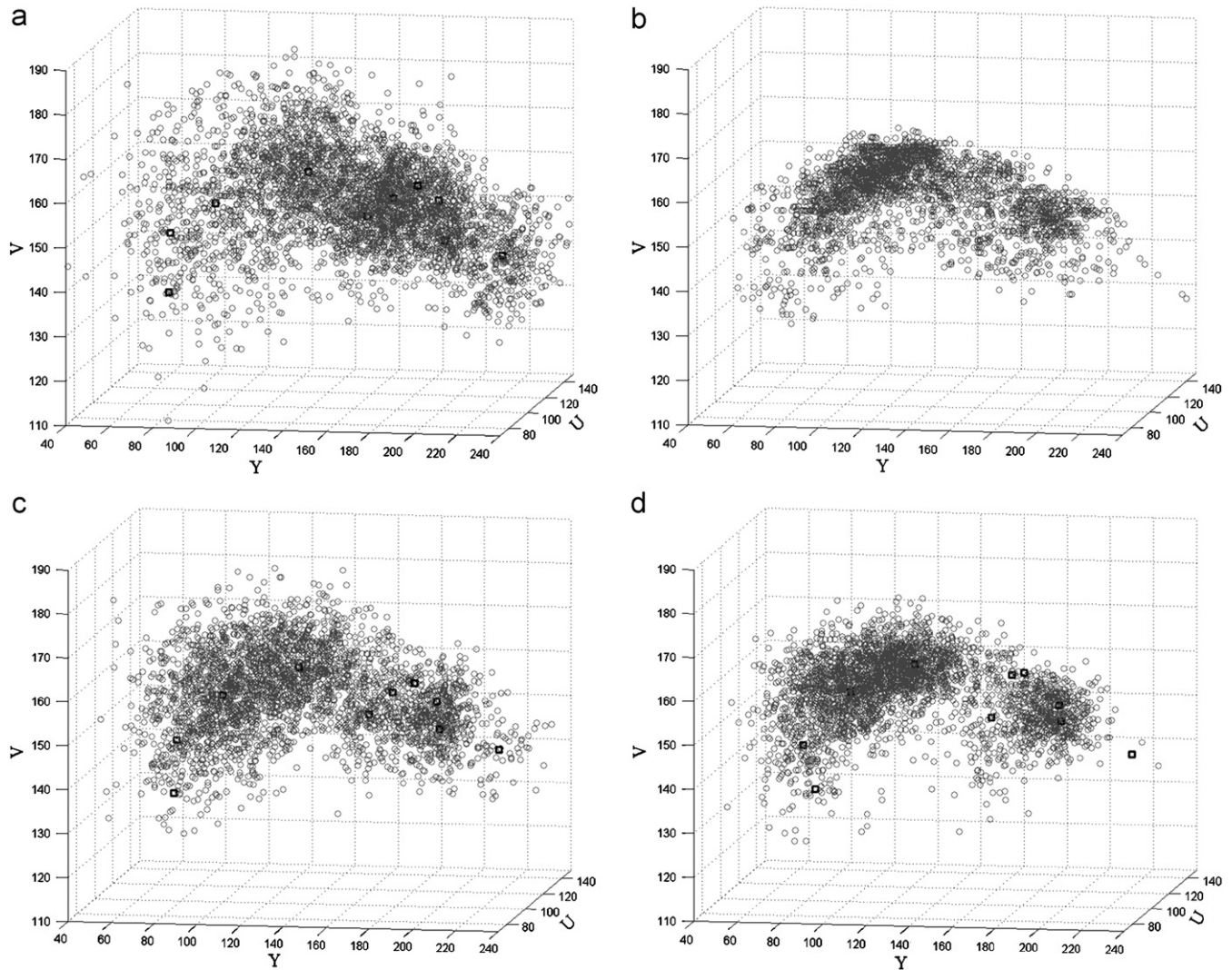


Fig. 5. Skin model evolution. (a), (c), and (d) are GMMs, where the squares denote the means of the Gaussians, and (b) plots the real skin pixels manually extracted from the image. (a) Genetic skin model $\Theta(0)$, (b) real skin pixels in an image, (c) skin model at the first iteration $\Theta(1)$, and (d) specific skin model $\Theta(3)$.

The recursive equations for updating GMM are based on the assumption that $P(m|\mathbf{x}, \Theta_m^{(N+1)}) \approx P(m|\mathbf{x}, \Theta_m^{(N)})$ (when N is large enough). The assumption indicates if the sample set is huge, the GMM fitted by N samples and the one fitted by $N + 1$ samples have almost the same posterior probabilities of \mathbf{x} . Eqs. (11), (12), and (16) are the recursive equations for updating the weight, the mean vector, and the covariance matrix of the m th Gaussian:

$$w_m^{(N+1)} \approx \frac{Nw_m^{(N)} + P(m|\mathbf{x}_{N+1}, \Theta_m^{(N)})}{N + 1}, \quad (11)$$

$$\boldsymbol{\mu}_m^{(N+1)} \approx \frac{Nw_m^{(N)} \boldsymbol{\mu}_m^{(N)} + \mathbf{x}_{N+1} P(m|\mathbf{x}_{N+1}, \Theta_m^{(N)})}{Nw_m^{(N)} + P(m|\mathbf{x}_{N+1}, \Theta_m^{(N)})}, \quad (12)$$

$$\begin{aligned} \Sigma_m^{(N+1)} &\approx \frac{\sum_{n=1}^N (\mathbf{x}_n - \boldsymbol{\mu}_m^{(N+1)})(\mathbf{x}_n - \boldsymbol{\mu}_m^{(N+1)})^T P(m|\mathbf{x}_n, \Theta_m^{(N)})}{Nw_m^{(N)} + P(m|\mathbf{x}_{N+1}, \Theta_m^{(N)})} \\ &+ \frac{(\mathbf{x}_{N+1} - \boldsymbol{\mu}_m^{(N+1)})(\mathbf{x}_{N+1} - \boldsymbol{\mu}_m^{(N+1)})^T P(m|\mathbf{x}_{N+1}, \Theta_m^{(N)})}{Nw_m^{(N)} + P(m|\mathbf{x}_{N+1}, \Theta_m^{(N)})}. \end{aligned} \quad (13)$$

Eq. (13) has not been the recursive form yet. To obtain the recursive equation for updating the covariance matrix, we have to resolve $\boldsymbol{\mu}^{(N+1)}$ (Eq. (12)) for an approximation:

$$\begin{aligned} \boldsymbol{\mu}_m^{(N+1)} &\approx \boldsymbol{\mu}_m^{(N)} + \frac{\mathbf{x}_{N+1} P(m|\mathbf{x}_{N+1}, \Theta_m^{(N)})}{Nw_m^{(N)} + P(m|\mathbf{x}_{N+1}, \Theta_m^{(N)})} \\ &= \boldsymbol{\mu}_m^{(N)} + \boldsymbol{\Gamma}_m^{(N+1)}. \end{aligned} \quad (14)$$

We denote the second term on the right-hand side of Eq. (14) by $\Gamma_m^{(N+1)}$. Substituting Eq. (14) into Eq. (13), the upper part of the first term on the right-hand side of Eq. (13) can be written as

$$\begin{aligned} & \sum_{n=1}^N (\mathbf{x}_n - \boldsymbol{\mu}_m^{(N+1)})(\mathbf{x}_n - \boldsymbol{\mu}_m^{(N+1)})^T P(m|\mathbf{x}_n, \Theta_m^{(N)}) \\ &= Nw_m^{(N)} \boldsymbol{\Sigma}_m^{(N)} + \sum_{n=1}^N \Gamma_m^{(N+1)} \Gamma_m^{(N+1)T} P(m|\mathbf{x}_n, \Theta_m^{(N)}). \end{aligned} \quad (15)$$

Substituting Eq. (15) into Eq. (13), then we can obtain the recursive equation for updating the covariance matrix:

$$\begin{aligned} & \boldsymbol{\Sigma}_m^{(N+1)} \\ & \approx \frac{Nw_m^{(N)} + \Gamma_m^{(N+1)} \Gamma_m^{(N+1)T}}{Nw_m^{(N)} + P(m|\mathbf{x}_{N+1}, \Theta_m^{(N)})} \\ & + \frac{(\mathbf{x}_{N+1} - \boldsymbol{\mu}_m^{(N+1)})(\mathbf{x}_{N+1} - \boldsymbol{\mu}_m^{(N+1)})^T P(m|\mathbf{x}_{N+1}, \Theta_m^{(N)})}{Nw_m^{(N)} + P(m|\mathbf{x}_{N+1}, \Theta_m^{(N)})}. \end{aligned} \quad (16)$$

Finally, the complete recursive equations for updating the skin model in each model-updating step can be summarized as the following loops: compute Eqs. (7), (11), (12) and (16) in turn for $N = \hat{K}, \dots, (\hat{K} + L - 1)$.

4.2. Analysis

The “generic to specific” evolution of the skin model is visualized via an example in Fig. 5 by employing the algorithm introduced in Section 4.1. Fig. 5(a) plots the generic skin model $\Theta(0)$, which is learned from a large collection of skin samples offline. Fig. 5(b) plots the real skin distribution (the skin pixels extracted manually from Fig. 1(a)). At the first iteration, $\Theta(0)$ is updated with the new skin pixels obtained in the sample-collection step, and it evolves to be $\Theta(1)$ (see Fig. 5(c)). One can see that $\Theta(1)$ looks a little similar to the real skin distribution. After three iterations, the skin model converges, and $\Theta(3)$ (see Fig. 5(d)), i.e., the SSM, is then able to accurately describe the real skin distribution.

To numerically compare the similarities between the skin models, $\Theta(n)$, $n = 0, 1, \dots$, and the real skin distribution, we provide the integrated absolute error (IAE) between them. IAE is defined as

$$\text{IAE} = \int_{\Omega} |p_{\Theta}(\mathbf{x}) - p_{\text{Real}}(\mathbf{x})| d\mathbf{x}, \quad (17)$$

where Ω denotes the entire integral space (YUV color space), $p_{\Theta}(\cdot)$ the mixture density function of the skin model Θ , and $p_{\text{Real}}(\cdot)$ the density in the real distribution. The range of IAE is $[0, 2]$ (0 for the two same distributions, 2 for the two distributions with no overlap). In Table 1, one can see that the IAE between $\Theta(0)$ and the real skin distribution is 1.094, which means little common information is shared; after three iterations, the IAE becomes 0.553 which indicates that $\Theta(3)$ is able to fit the real distribution much better.

Table 1
Integrated absolute error

| Skin model at the n th iteration | IAE |
|------------------------------------|-------|
| $\Theta(0)$ generic skin model | 1.094 |
| $\Theta(1)$ | 0.769 |
| $\Theta(2)$ | 0.561 |
| $\Theta(3)$ specific skin model | 0.553 |

5. Boundary potential field

After incremental learning iterations, the skin map generated by the SSM at the final iteration can be obtained. It is the most accurate skin map and can be outputted directly as the final segmentation result. However, an additional post-processing could be further performed to fill up the remaining interstices on the skin map (e.g., the enclosed regions in Fig. 6(b)). The remaining interstices which have not been addressed by incremental learning always get abrupt color transitions, color-based region growing may fail. Thus, we build the boundary potential field, which is able to assign lower energies to the interstices, outside the skin regions to guide region growing to fill up the interstices.

In the Post-process, region growing is coactivated by both color similarity (D) and potential energy (E):

$$D \times E \leq \text{Constant}. \quad (18)$$

Eq. (18) indicates the region growing condition, i.e., if the product of the two factors is small enough, the region could continue to grow. We cannot merely depend on either factor, because (1) if we only consider D , the background may be obtained since it may get the similar colors to the skins, but this can be avoided by employing E since E is large in the background; and (2) if we only consider E , the small coverings on the body may be obtained since they may get lower potential energies, but this can be avoided by employing D since D is large on the coverings.

We begin to define the potential energy. Firstly, three pixel sets should be defined: $A = \{\text{the pixels on the skin map}\}$, $V = \{\text{the pixels in the skin regions}\}$, and $U = A - V$. For each element \mathbf{z} in U , an $h \times h$ window \hat{W} is located at it. By constructing the 2-D coordinate system centered at \mathbf{z} , the potential energy at \mathbf{z} is defined as

$$E(\mathbf{z}) = \sum_{x=-h/2}^{h/2} \sum_{y=-h/2}^{h/2} \frac{1}{\sqrt{x^2 + y^2}}, \quad (x, y) \in \hat{W} \cap V. \quad (19)$$

For the comparability between D and E , E should be normalized to be in the range of $[0, 255]$:

$$E(\hat{W}) = \sum_{x=-h/2}^{h/2} \sum_{y=-h/2}^{h/2} \frac{1}{\sqrt{x^2 + y^2}}, \quad (x, y) \in \hat{W}, \quad (20)$$

$$\tilde{E}(\mathbf{z}) = 255 \exp\{-E(\mathbf{z})/E(\hat{W})\}. \quad (21)$$

Fig. 6(c) illustrates the boundary potential field on the skin map denoted by gray scales, and the deeper gray regions indicate the area where lower potential energies are assigned.

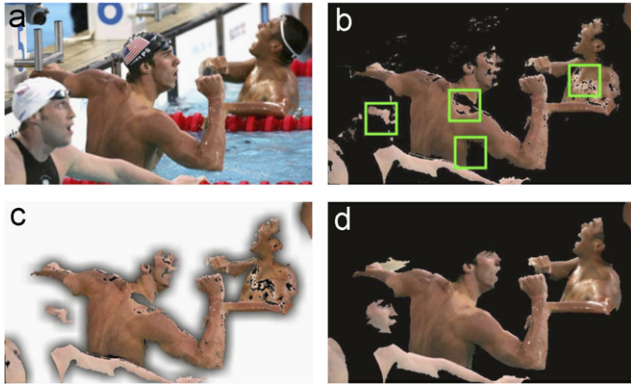


Fig. 6. Filling up the remaining interstices. The boundary potential field is denoted by gray areas in (c). (a) Original image, (b) skin map (second iteration), (c) boundary potential field, and (d) final skin map.

The region growing algorithm in the Post-process is as same as Algorithm 1 in the Incremental-process except for the conditional expression. The “edge” information is not considered but the potential energies are introduced; thus the conditional expression (at the seventh line) becomes $D_z \times E_z \leq \delta$ ($\delta = 1000$). For \mathbf{z} in the SSL, E_z and D_z are defined as

$$E_z = \frac{\sum_i \tilde{E}(\mathbf{z})}{|W \cap U|}, \quad \mathbf{z} \in W \cap U, \quad (22)$$

$$D_z = \frac{\|\bar{\mathbf{x}} - \mathbf{x}\|}{\sqrt{3}}, \quad (23)$$

where $|\cdot|$ denotes the number of elements in the set, and W , $\bar{\mathbf{x}}$, and \mathbf{x} are the same as the ones defined in Algorithm 1. Fig. 6(d) illustrates the final skin map after filling up the interstices.

6. Experimental results

We have performed a series of experiments to select the optimal values for the parameters and evaluate the performance of the proposed framework by comparing it with the performance of Ref. [5]. Several well-known face databases, e.g., PIE database [20] and Physics-based face database [21], have been adopted for color-based skin pixels detection. However, from the point of view of our objective, these databases are not suitable for testing our method since these images only contain the faces located in the same scene. In particular, our method aims to segment integrated exposed regions on the human bodies including faces. Therefore, we constructed a real-life image database for the experiments, in which 1624 color images were collected from the Web. The images in the database are greatly diverse in terms of imaging conditions, illumination conditions, shooting angles, scenes, and races (e.g., Caucasians, Asians, and Africans). We randomly selected 1000 images as the test set, and used the remaining 624 images as the training set. The statistics of image categories in the test set is listed in Table 2.

First of all, we manually annotated the accurate skin maps (e.g., Fig. 1(b)) as the ground truth (GT) for all the images in the database. Then, we sampled 18 million skin pixels from the

Table 2
Test set

| Category | Number |
|-----------------------------------|--------|
| Advertisements/posters | 58 |
| Daily life photos | 208 |
| Models/fashions/Miss world | 211 |
| Portraits/half-length photos | 156 |
| Sports: Aquatics/gymnastics | 197 |
| Sports: Athletics/wrestling, etc. | 125 |
| Sports: Tennis/volleyball, etc. | 45 |
| Total | 1000 |

Table 3

Image scales

| Image size (if aspect ratio is 3:2) | W | \hat{W} |
|-------------------------------------|---------|-----------|
| 24,576 (216 × 144) | 18 × 18 | 9 × 9 |
| 55,296 (288 × 192) | 24 × 24 | 12 × 12 |
| 98,304 (384 × 256) | 32 × 32 | 16 × 16 |
| 153,600 (480 × 320) | 40 × 40 | 20 × 20 |

GT skin maps in the training set, and used them to train the generic skin model (i.e., $\Theta(0)$) offline.

In the experiments, we evaluated the individual segmentation result based on the true positive rate and the false positive rate, which were calculated by pixel-wise comparing the generated skin map with the GT skin map, and evaluated the average performance on the whole test set by averaging the 1000 individual results. All the experiments were performed on a 3.0 GHz CPU.

6.1. Parameters selection

Three important parameters in the proposed framework which may effect on the performance are required to be selected, i.e., the probability threshold (τ) for the skin model, the learning rate (λ), and the image size. We selected the optimal values for the first two parameters by adjusting τ in the set $\{2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^0, 2^1, 2^2, 2^3, 2^4\} \times 10^{-5}$ and λ in the set $\{0.3, 0.4, 0.5, 0.6, 0.7\}$. For the image size, we defined four scales in Table 3, where W is the region growing window defined in Section 3.3 and \hat{W} is the potential field window defined in Section 5. All the test images and the corresponding GT skin maps were scaled up/down to fit the four image scales. Thus, we got 180 combinations of the three parameters. For each combination, we performed a run of skin region segmentation on the test set and calculated the average performance. We selected the optimal values for the parameters by investigating the true positive rate and the false positive rate curves.

Figs. 7 and 8 plot the average true positive rate and the average false positive rate curves, respectively, with respect to τ , λ , and the image size. The probability threshold τ directly effects on the segmentation results. The false positive rates gradually reduce as τ increases; however, the true positive rates do not increase correspondingly with τ , since if the threshold is loose enough, more non-skin pixels would be collected by

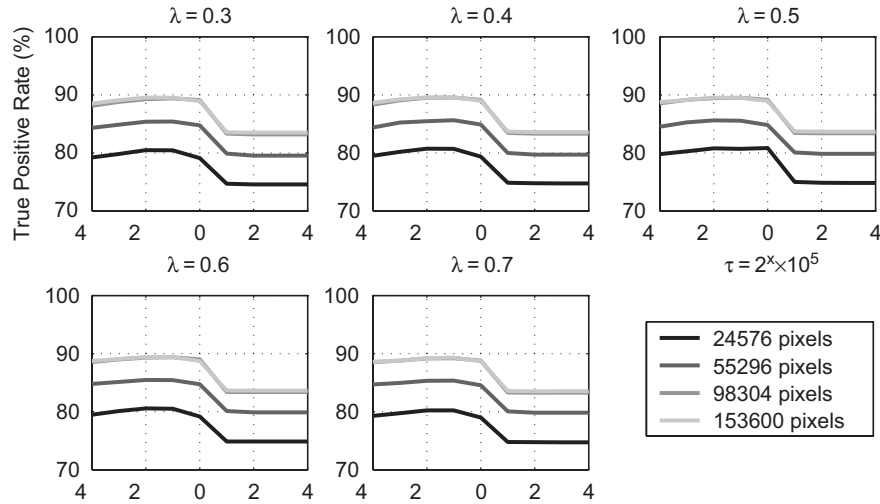


Fig. 7. Average true positive rate.

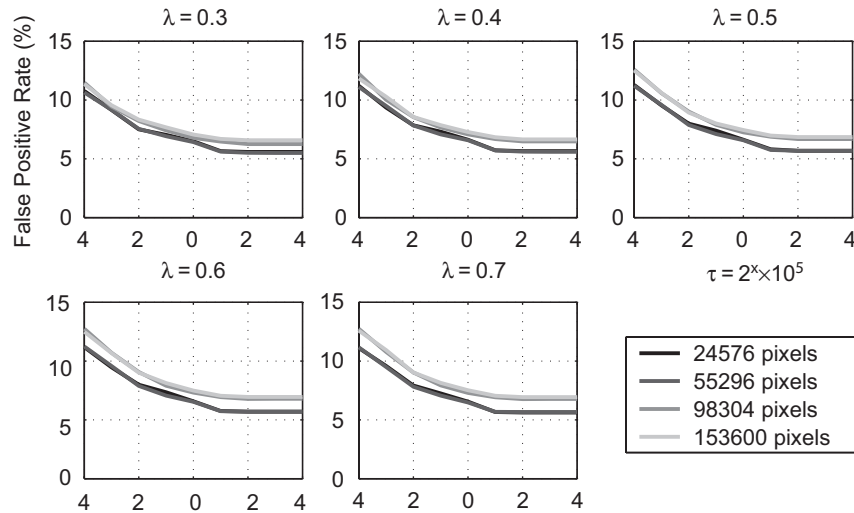


Fig. 8. Average false positive rate.

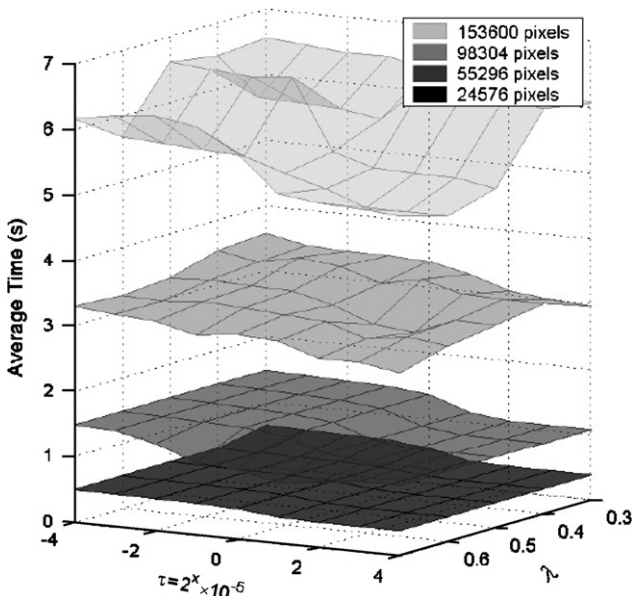


Fig. 9. Average time statistics.

Table 4

The average performances with the optimal values for τ and λ

| Image size | TPR (%) | FPR (%) | Average time (s) |
|------------|---------|---------|------------------|
| 24,576 | 6.62 | 80.85 | 0.462 |
| 55,296 | 6.58 | 84.81 | 1.367 |
| 98,304 | 7.25 | 89.13 | 3.145 |
| 153,600 | 7.44 | 88.93 | 5.485 |

region growing for updating the skin model, and the skin model would have few skin color components after several iterations to detect skin pixels. In both Figs. 7 and 8, the five subplots are similar, which indicates that the learning rate λ does not observably effect on the segmentation results. However, in Fig. 9, one can see that the average time for processing an image reduces gradually with the growth of λ , since a large λ may accelerate the convergence of the skin model. One can also see that the image size slightly influences the average false positive rate, but greatly influences the average true positive rate. The reason of this phenomenon is that, in the small images, the trivial skin

regions can be hardly obtained by region growing, thus the true positive rates are lower. On the other hand, the true positive rates cannot be unlimitedly improved by enlarging the image. When the image size exceeds a critical scale (e.g., 98,304), the increasing speed of the true positive rates will slow down

and stop (e.g., the curves of 153,600 and the curves of 98,304 in Fig. 7 are almost overlapped).

Based on the results in Figs. 7 and 8, the optimal value for the threshold τ should be 10^{-5} by balancing the true positive rate and the false positive rate. For the learning rate λ , we set $\lambda = 0.5$ for a steady convergence rate. After setting the optimal values for τ and λ , we list the average performances in Table 4 with respect to the image size. To achieve the best tradeoff between the segmentation result and the processing time, the images which comprise about 50,000–100,000 pixels can be adopted in practice.

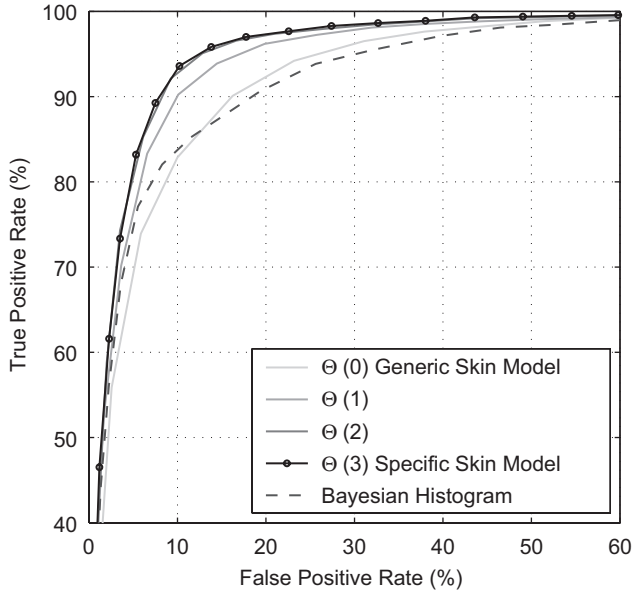


Fig. 10. ROC curves.

6.2. Performance evaluation

Phung et al. [2] concluded that Bayesian classifier with histograms proposed by Ref. [5] performed best among the state-of-the-art pixel-based skin segmentation methods. Since most of the reported adaptive methods, e.g., Refs. [7,10,11], are based on various additional preconditions and assumptions, they can only work in certain applications and can be hardly implemented for skin region segmentation in any images as a general method. Thus, we only chose the well-known Bayesian histogram [5] for comparing with the SSM generated by the proposed framework, and demonstrated that the SSM is more robust than the static skin models.

For the comparability between the two methods, the post-processing (i.e., the dashed rectangles in Fig. 3) in the proposed



Fig. 11. Examples. The region growing results are denoted by shaded patches in column (c) and (e), and the boundary potential fields are denoted by gray areas in column (g). (a) Original images, (b) initial skin maps, (c) region growing (first iteration), (d) skin maps (first iteration), (e) region growing (second iteration), (f) skin maps (second iteration), (g) boundary potential field, and (h) final skin maps.

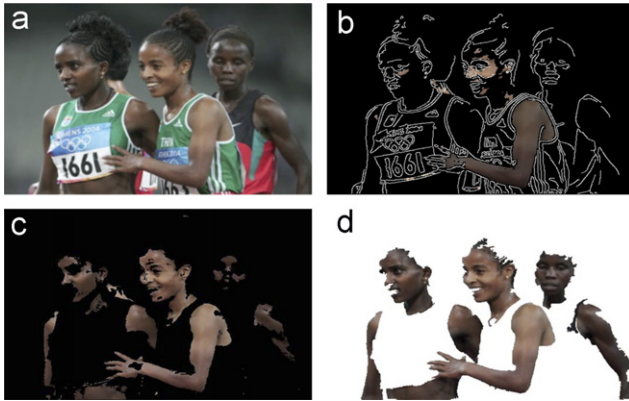


Fig. 12. An example of the blackish people. (a) Original image, (b) region growing (first iteration), (c) skin map (second iteration), and (d) final skin map.

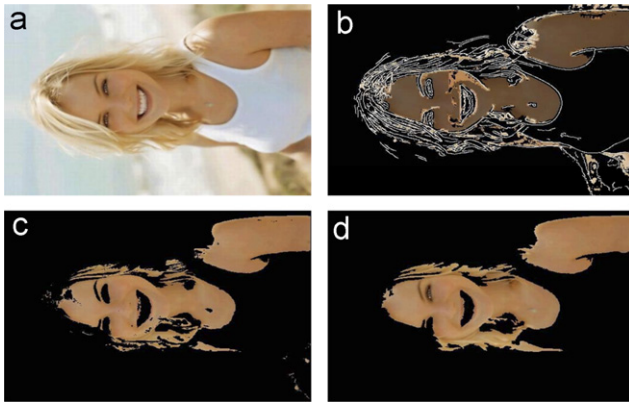


Fig. 13. An example of the blonde. (a) Original image, (b) region growing (first iteration), (c) skin map (second iteration), and (d) final skin map.

framework is omitted in this experiment. We also used the same training set to construct the positive and the negative skin histograms for the Bayesian histogram. To compare the average performances of the SSM and the Bayesian histogram on the test set (image size 98,304), we plotted the ROC curves (see Fig. 10) for the two methods by tuning the thresholds. For our method, we plotted the ROC curves for the skin models $\Theta(n)$, $0 \leq n \leq 3$, at each incremental learning iteration, where $\Theta(0)$ is the generic skin model which can be viewed as the baseline and $\Theta(3)$ is the SSM. By investigating the ROC curves, one can see that the SSM is able to observably improve both the true positive rates and the false positive rates as we expected in Section 2; one can also see that $\Theta(2)$ has already converged, thus the iterative time can be fixed to be 2 in practice to further reduce the computational costs.

The SSM outperformed the Bayesian histogram due to the advantage of being able to incrementally learn the specific skin color information from an image. In Fig. 11, some examples are illustrated and one can see intuitively how the proposed framework learns the specific skin color information from an image step by step, and finally generates the accurate skin maps. Moreover, the proposed framework is also robust for blackish people (see Fig. 12) and blondes (see Fig. 13).

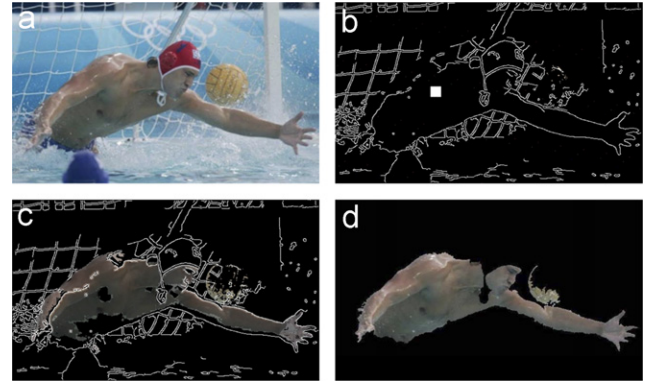


Fig. 14. Interactive mode. (a) Original image, (b) assign a PSR manually, (c) region growing (first iteration), and (d) final skin map.

6.3. Interactive mode

Since the proposed method does not aim at correcting the white balance in the image, if the image gets an incorrect white balance, the generic skin model $\Theta(0)$ may sometimes fail in detecting the PSRs. However, one can deal with such cases in the interactive mode, i.e., assigning a PSR manually in the Pre-process. We take Fig. 14(a), for example, to illustrate the interactive mode. Since the bluish ambient light influences the image, the entire skin surface reflects the incorrect color and the generic skin model failed to detect any PSR. Thus, we manually label a PSR for the image (see Fig. 14(b)), and the remaining steps can be performed automatically as usual.

7. Conclusion

We have proposed an incremental learning framework for accurate skin region segmentation in real-life color images. The proposed framework follows a “generic to specific” evolution, in which the skin model is updated with the skin pixels collected by region growing, to learn the specific skin model (SSM) for the test image in real-time. Our method is able to automatically adapt to the skin color variations induced by illumination conditions and inherent skin colors, thus it can be applied to any color images without limitations. Our method is different from the reported adaptive skin detection methods in the following aspects: (1) our method aims to segment arbitrary exposed parts on the human bodies; (2) our method can adapt to the skin color variations in a certain image; and (3) our method has a general framework, which does not depend on additional preconditions and assumptions, thus it would not be restricted in a certain application scenario.

Due to the SSM, our method can outperform those static skin detectors, e.g., the well-known Bayesian histogram, on a large real-life image database. The experimental results have justified our motivation, i.e., to improve both the true positive rate and the false positive rate for the test image by learning the skin color information from itself.

The proposed framework focuses on the accuracy and integrity of the generated skin map, thus it is very suitable for

the applications such as gesture recognition and objectionable image filtering. Moreover, the proposed incremental learning framework also can be applied to region segmentation for arbitrary objects (e.g., sky, vegetation, mountain, etc. in image retrieval systems).

References

- [1] V. Vezhnevets, V. Sazonov, A. Andreeva, A survey on pixel-based skin color detection techniques, in: Proceedings of the Graphicon-2003, Moscow, Russia, 2003, pp. 85–92.
- [2] S.L. Phung, A. Bouzerdoum, D. Chai, Skin segmentation using color pixel classification: analysis and comparison, *IEEE Trans. Pattern Anal. Mach. Intell.* 27 (1) (2005) 148–154.
- [3] E.H. Land, The retinex theory of color vision, *Sci. Am.* 237 (6) (1977) 108–128.
- [4] B. Martinkauppi, M. Soriano, M. Pietikainen, Detection of skin color under changing illumination: a comparative study, in: Proceedings of the 12th International Conference on Image Analysis and Processing (ICIAP'03), 2003, pp. 652–657.
- [5] M.J. Jones, J.M. Rehg, Statistical color models with application to skin detection, in: Proceedings of the Computer Vision and Pattern Recognition, 1999, pp. 274–280.
- [6] R.-L. Hsu, A.-M. Mohamed, A.K. Jain, Face detection in color images, *IEEE Trans. Pattern Anal. Mach. Intell.* 24 (5) (2002) 696–706.
- [7] M. Soriano, B. Martinkauppi, S. Huovinen, M. Laaksonen, Adaptive skin color modeling using the skin locus for selecting training pixels, *Pattern Recognition* 36 (3) (2003) 681–690.
- [8] Y. Raja, S.J. McKenna, S. Gong, Tracking and segmenting people in varying lighting conditions using colour, in: Proceedings of the Automatic Face and Gesture Recognition, Nara, Japan, 1998.
- [9] L. Sigal, S. Sclaroff, V. Athitsos, Estimation and prediction of evolving color distributions for skin segmentation under varying illumination, in: Proceedings of the Computer Vision and Pattern Recognition, 2000, pp. 152–159.
- [10] H. Sahbi, N. Boujemaa, From coarse to fine skin and face detection, in: Proceedings of the ACM International Conference on Multimedia (MM'00), 2000, pp. 432–434.
- [11] Q. Zhu, C.-T. Wu, K.-T. Cheng, Y.-L. Wu, An adaptive skin model and its application to objectionable image filtering, in: Proceedings of the ACM International Conference on Multimedia (MM'04), 2004, pp. 56–63.
- [12] A. Albiol, L. Torres, E.J. Delp, Optimum color spaces for skin detection, in: Proceedings of the IEEE International Conference on Image Processing (ICIP'01), 2001, pp. 122–124.
- [13] A.P. Dempster, N. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, *J. R. Stat. Soc. Ser. B (Methodol.)* 1 (39) (1977) 1–38.
- [14] J. Canny, A computational approach to edge detection, *IEEE Trans. Pattern Anal. Mach. Intell.* 8 (6) (1986) 679–698.
- [15] S.J. Roberts, D. Husmeier, I. Rezek, W. Penny, Bayesian approaches to Gaussian mixture modeling, *IEEE Trans. Pattern Anal. Mach. Intell.* 20 (11) (1998) 1133–1142.
- [16] R. Adams, L. Bischof, Seeded region growing, *IEEE Trans. Pattern Anal. Mach. Intell.* 16 (6) (1994) 641–647.
- [17] P. Hall, Y. Hicks, A method to add Gaussian mixture models, Technical Report 2004-03, 2004.
- [18] Z. Zivkovic, F. Heijden, Recursive unsupervised learning of finite mixture models, *IEEE Trans. Pattern Anal. Mach. Intell.* 26 (5) (2004) 651–656.
- [19] R. Neal, G. Hinton, A view of the EM algorithm that justifies incremental, sparse, and other variants, in: M.I. Jordan (Ed.), *Learning in Graphical Models*, Kluwer, Dordrecht, 1998.
- [20] T. Sim, S. Baker, M. Bsat, The CMU pose, illumination, and expression (PIE) database, in: Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition (AFGR'02), 2002.
- [21] B. Martinkauppi, Face colour under varying illumination—analysis and applications, Ph.D. Dissertation, University of Oulu, 2002.

About the Author—BIN LI received his B.Eng. degree in software engineering from Southeast University, Nanjing, China, in 2004. He is now a Ph.D. student with the Department of Computer Science and Engineering, Fudan University, Shanghai, China. His current research interests include statistical machine learning and its application to multimedia information retrieval.

About the Author—XIANGYANG XUE received his B.S., M.S., and Ph.D. degrees in communication engineering from Xidian University, Xi'an, China, in 1989, 1992, and 1995, respectively. Since 1995, he has been with the Department of Computer Science and Engineering, Fudan University, China, where he is currently a professor. His research interests include multimedia information processing and retrieval.

About the Author—JIANPING FAN received his M.S. degree in theory physics from Northwestern University, Xi'an, China, in 1994, and the Ph.D. degree in optical storage and computer science from Shanghai Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, Shanghai, China, in 1997. He was a Researcher at Fudan University, Shanghai, China, during 1998. From 1998 to 1999, he was a Researcher with the Japan Society for Promotion of Sciences (JSPS), Department of Information System Engineering, Osaka University, Osaka, Japan. From September 1999 to 2001, he was a Researcher in the Department of Computer Science, Purdue University, West Lafayette, IN. He is now an Associate Professor in the Department of Computer Science, University of North Carolina, Charlotte. His research interests include nonlinear systems, error correction codes, image processing, video coding, semantic video computing, and content-based video indexing and retrieval.