Contribution of Skin Color Cue in Face Detection Applications

Dohyoung Lee, Jeaff Wang and Konstantinos N. Plataniotis

Abstract Face detection has been considered as one of the most active areas of research due to its wide range of applications in computer vision and digital image processing technology. In order to build a robust face detection system, several cues, such as motion, shape, color, and texture have been considered. Among available cues, color is one of the most effective ones due to its computational efficiency, high discriminative power, as well as robustness against geometrical transform. This chapter investigates the role of skin color cue in automatic face detection systems. General overview of existing face detection techniques and skin pixel classification solutions are provided. Further, illumination adaptation strategies for skin color detection are discussed to overcome the sensitivity of skin color analysis against illumination variation. Finally, two case studies are presented to provide more realistic view of contribution of skin color cue in face detection frameworks.

1 Introduction

Face detection refers to a task of processing image/video data in order to determine the position/scale of any faces in them [31]. With recent advancement in digital imaging devices and multimedia technologies, face detection has received a significant deal of attention from the research communities due to its wide range of applications from video surveillance system, human-computer interface (HCI), to face image database

D. Lee (✉) · J. Wang · K. N. Plataniotis
Multimedia Lab, The Edward S. Rogers Department of Electrical and Computer Engineering,
University of Toronto, 10 King's College Road, Toronto, Canada
e-mail: dohyoung.lee@utoronto.ca

J. Wang
e-mail: jeaff.wang@utoronto.ca

K. N. Plataniotis
e-mail: kostas@comm.utoronto.ca

M. E. Celebi and B. Smolka (eds.), *Advances in Low-Level Color Image Processing*, Lecture Notes in Computational Vision and Biomechanics 11,
DOI: 10.1007/978-94-007-7584-8_12, © Springer Science+Business Media Dordrecht 2014

management. For instance, a successful solution to the face detection problem is potentially useful for user scenarios such as:

1. Surveillance cameras have been widely deployed in many strategic places to ensure public safety by monitoring criminal and terrorist activities. As a result, efforts have been made on the implementation of an intelligent system that performs monitoring task automatically without human intervention. Such task requires a preliminary operation that locates faces within the video scene, followed by other surveillance operations, such as face tracking and gait recognition.
2. As mobile phones become increasingly powerful in terms of processing resource and data capacity, security of the data stored in them such as contact information and confidential documents becomes very important. The conventional way of protecting such sensitive data is to have them password protected. However, latest mobile phones started to offer face recognition technology as a more secure and convenient security means since it provides distinctive print to get access while the user does not need to memorize passwords. To this end, an user can take a still image of himself/herself for identification through the image sensor. The detection of face region in captured image data is an essential initialization stage to complete user verification.

The importance of face detection cannot be overemphasized in the aforementioned applications since they operate under an assumption that faces are already accurately located in the given image or in the image sequence prior to main processing operation. In the literature, numerous strategies have been proposed; however, face detection is still considered to be challenging since faces are non-rigid objects and vary substantially in shape, color, and texture [75]. Key technical challenges involved in face detection problem include:

- **Pose Variation**: face object varies significantly depending on its relative position between camera-face (from frontal views to different angles of side and tilt views)
- **Imaging Condition**: the variation of illumination condition (e.g. color temperature of light source) and camera characteristic (e.g. sensor spectral sensitivity) affects the appearance of color or texture of face region.
- **Occlusions**: complete face region might not be fully visible to camera, and some facial feature might be hidden by other object

Progress has been made to solve these problems by using different aspects of facial characteristics including geometric relationships between facial features (e.g. eyes, nose, and mouth), skin color, facial texture patterns, and so forth. Although many different cues have been utilized in the face detection system, skin color cue is one of the most effective ones due to its robustness towards geometric changes (e.g. scaling and rotation) and computational efficiency.

In this chapter, we investigate the role of skin color cue in automatic face detection systems. First we provide general overview of face detection techniques with emphasis on use of color information (Sect. 2). In Sect. 3, we provide review of various color representation and skin pixel classification methodologies based on color information. In addition, illumination adaptation strategies for skin color detection is discussed. In Sects. 4 and 5, we introduce two case studies of skin color cue usage in face detection framework. By presenting two distinct use cases of color information in facial analysis, the effectiveness of color cue in terms of detection performance and computational efficiency is emphasized. Additionally, the influence of stable color representation on face detection performance is addressed by applying color constancy algorithms prior to skin color detection process. Finally, conclusions are drawn in Sect. 6.

2 Face Detection and Color Cue

Face detection problem has received tremendous research interests since the 1970s due to its importance in computer vision applications such as human identification and tracking, human-computer interaction, and content-based image retrieval. For instance, the recognition of a face in visual data typically involves three main stages (Fig. 1), where the detection of face region is an essential initialization step for face alignment (i.e. registration) that any subsequent operations are directly influenced by its accuracy [81].

The purpose of face detection is to process still images or image sequences to find the location(s) and the size(s) of any faces in them [31]. Although face detection is a trivial task for humans, it is very challenging to build a stable automatic solution since face patterns significantly vary under different facial poses/expressions, occlusion conditions, and imaging condition (e.g. illumination condition, sensor characteristics). Various face detection algorithms have been proposed [31, 75, 79], but achieving highly accurate detection performance while maintaining reasonable computational costs still remains to be a challenging issue.

In general, existing face detection methods are grouped into the two main categories [31]: (i) *feature-based approach*, (ii) *image-based approach*. In feature-based approach [33, 48, 54, 58, 72], explicit face features (e.g. eyes, mouth, nose and face contour) are extracted, then relationships between them (such as geometric and morphologic relationships) are used to determine the existence of the face. For instance, Sobottka and Pitas [72] proposed a two-stage face detection framework to locate face regions in color image. The first stage is dedicated to segment face-like regions by skin color analysis using hue and saturation information in HSV colorspace, followed by shape analysis using ellipse fitting. Afterwards, grayscale information of detected face-like regions are examined to verify them by locating eye and mouth features.

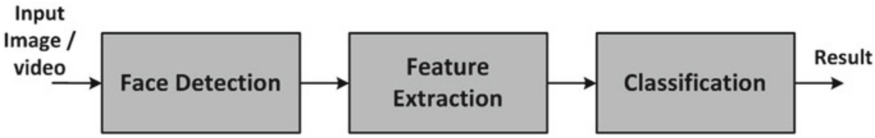

Fig. 1 Block diagram of a generic face recognition system

Extending the feature-based approach, Hsu et al. [33] localized face candidates from color image using skin color cue in YCbCr colorspace and constructed eye, mouth, face boundary maps to verify each face candidate. The methods in this category are advantageous due to their relatively simple implementation and high detection accuracy in uncluttered backgrounds. In particular, skin color cue is exceptionally popular and successful in feature-based approach due to its simplicity and high discrimination power. Some of these methods remain very popular nowadays in certain applications such as mobile phone applications. However, they tend to have difficulties in dealing with challenging imaging conditions such as varying illumination and complex background, as well as low resolution images containing multiple faces.

Alternatively, image-based approach [9, 13, 67, 80] uses machine learning techniques to capture unique and implicit face features, treating the face detection problem as a binary classification problem to discriminate between face and non-face. Often, methods in this category require tremendous amount of time and training data to construct a stable face detection system. However, in recent years, the rapid advancement in digital data storage and digital computing resources has made the image-based approaches feasible to many real-life applications and they become extremely popular due to their enhanced robustness and superior performance against challenging conditions compared to feature-based approaches.

One of the most representative works in image-based approach is the Viola-Jones's face detection framework [67], a Haar-like feature based frontal face detection system for grayscale images. The Haar-like feature represents the differences in grayscale between two or more adjacent rectangular regions in the image, characterizing local texture information. In Viola-Jones framework, AdaBoost learning algorithm is used to handle following three fundamental problems: (i) learning effective features from a large Haar-like feature set, (ii) constructing weak classifiers, each of which is based on one of the selected features, (iii) boosting the weak classifiers to construct a strong classifier. The authors applied the integral image technique for fast computation of Haar-like features under varying scale and location, achieving real-time operation[1]. However, the simplicity of Haar-like features in Viola-Jones face detector causes limited performance under many complications, such as illumination variation. More recent approaches address this problem by using an alternative texture feature called Local Binary Pattern (LBP), which is introduced by Ojala et al. [50] to offer enhanced discriminative power and tolerance towards illumination variation. The LBP descriptor encodes the relative gray value differences from a 3×3 neighborhood of an image patch as demonstrated in Fig. 2. By taking the center pixel as a threshold value, every neighboring pixel is compared against the threshold to produce a 8-bit string, and then binary string is converted to decimal label according to assigned weights. Many variants of LBP features are considered thereafter, such as LBP Histogram [26], Improved Local Binary Pattern (ILBP) [35], Multi-Block LBP (MB-LBP) [80], and Co-occurrence of LBP (CoLBP) [47].

[1] The term real-time implies the capability to process image frames with a rate close to the examined sequence frame rate. In [67], real-time requirement is defined to be approximately 15 frames per second for 384x288 image.

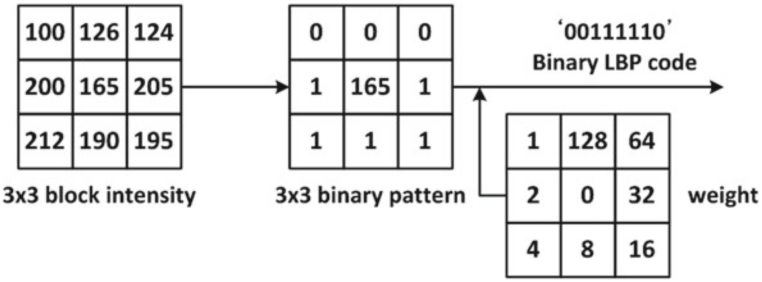

Fig. 2 Example of basic LBP feature computation

Although considerable progress has been made in aforementioned image-based face detection methodologies, the main emphasis has been placed on exploiting various grayscale texture patterns. A few recent researches [12, 25] have shown that color cue could provide complementary information to further enhance the performance of image-based approaches. Specifically, color information could potentially enhance the performance in following two aspects: (i) using skin color information, one may effectively reduce the search regions for potential face candidates by identifying skin color regions and performing subsequent texture analysis on detected skin region only, hence avoiding heavy computations caused by exhaustive scan of the entire image, (ii) color is a pixel-based cue which can be processed regardless of spatial arrangement, hence, it offers computational efficiency as well as robustness against geometric transformation such as scaling, rotation, and partial occlusion.

Overall, color can be applied in face detection systems, either as a primary or a complementary feature to locate faces along with shape, texture, and motion. In feature-based approach, which is suitable for resource constrained systems, color provides visual cue to focus attention in the scene by identifying a set of skin-colored regions that may contain face objects. It is followed by subsequent feature analysis where each skin-colored region is analyzed using facial geometry information. In image-based approach, faster and more accurate exhaustive face search can be achieved by using skin color modality with the purpose of guiding the search. Typical examples of each face detection approach and use of color cue are described in Sects. 4 and 5. Since accurate classification between skin and non-skin pixel is a key element for reliable implementation of face detection systems, in Sect. 3, we provide extensive review of existing skin color classification methodologies.

3 Skin Color Detection

Color is an effective cue for identifying regions of interest/objects in the image data. For visual contents of natural scene, a widely accepted assumption is that the color corresponds to a few categories have the most perceptual impact on the human visual system [76]. Researches indicate that skin tones, blue sky, and green foliage constitute such basic classes and belong to a group of color termed *memory colors*. Among memory colors, skin color has been regarded as the most important ones due to its importance in human social interaction and its dominant usage in image/video analysis. The application of skin color analysis is not only limited to face detection system, which is the main focus of this chapter, but also includes content-based image retrieval system, human-computer interaction domain, and memory color enhancement system.

Skin color analysis in face detection framework involves a pixel-wise classification to discriminate skin and non-skin pixels in color images. Therefore, skin color detection process can be seen as a binary classification problem that a certain color pixel $\mathbf{c} = [c_1, c_2, c_3]^T$ is mapped to an output label $y \in \{w_s, w_n\}$, where w_s and w_n represent skin and non-skin classes respectively.

Detection of skin color is considerably challenging not only because it is sensitive to varying illumination conditions and camera characteristics, but also it should be able to handle individual differences caused by ethnicity, age, and gender. Skin color detection involves two important sub-problems: (i) selection of a suitable color representation to perform classification (discussed in Sect. 3.1), (ii) selection of modeling scheme to represent skin color distribution (discussed in Sect. 3.2).

3.1 Color Representation

An appropriate representation of color signal is crucial in detection of skin pixels. An ideal colorspace for skin color analysis is assumed to: (i) minimize the overlap between skin and non-skin color distributions in the given colorspace, (ii) provide robustness against varying illumination condition, (iii) provide separability between luminance and chrominance information. Several colorspaces, such as RGB [36, 61], normalized RGB [4, 6, 23, 59], YCbCr [7, 29, 33, 62], HSV [30, 49, 55], CIELAB [78] and CIELUV [74], have been used in skin color detection. In this section, we will focus on most widely used colorspaces in the image processing research, including RGB, normalized RGB, YCbCr, and HSV.

3.1.1 RGB and Normalized RGB

RGB is a fundamental way of representing color signal which is originated from cathode ray tube (CRT) display. The RGB model (Fig. 3) is represented by a 3-dimensional cube with R, G, and B at the corners on each axis. RGB is most dominantly used representation for processing and storing of digital image data. Therefore, it has been used in many skin color detection researches [36, 51, 61]. However, its poor separability between luminance and chromaticity information, and its highly sensitive nature against illumination variation are main limitations for skin color analysis purpose. Rather than original RGB format, its normalized variant is consid-

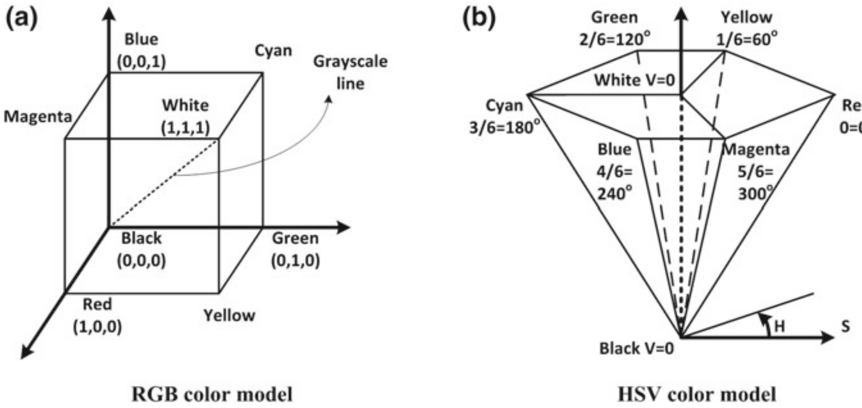

Fig. 3 RGB cube and HSV hexcone representations

ered to be more robust in skin color detection since it reduces the dependency of each component to illumination changes[2] [38]. Normalized RGB representation can be obtained by normalizing RGB values by their intensity ($I = R + G + B$):

$$r = \frac{R}{R+G+B}, \; g = \frac{G}{R+G+B}, \; b = \frac{B}{R+G+B} \tag{1}$$

where each RGB primaries are given in linear RGB[3]. Because $r + g + b = 1$, no information is lost if only two elements are considered.

3.1.2 YCbCr

YCbCr is the Rec. 601 international standard (see 2) for studio quality component digital video. In YCbCr, color is represented by luma(Y), computed as a weighted sum of nonlinear (gamma-corrected) RGB ,and two chroma components Cr and Cb that are formed by subtracting luma value from R and B components. The Rec. 601 specifies 8 bit (i.e. 0–255) coding of YCbCr. All three Y, Cb, and Cr components have reserved ranges to provide footroom and headroom for signal processing as follows:

[2] Under the white illumination condition with Lambertian reflection assumption, normalized RGB is invariant to illumination direction and illumination intensity [21]

[3] Linear RGB implies that it is linear to the physical intensity, whereas nonlinear RGB is non-linear to intensity. Such nonlinearity is introduced to RGB signal by gamma correction process in order to compensate a nonlinear response of CRT display devices.

$$\begin{cases} Y' &= 16 + (65.738R' + 129.057G' + 25.064B') \\ Cb &= 128 + (-37.945R' - 74.494G' + 112.439B') \\ Cr &= 128 + (112.439R' - 94.154G' - 18.285B') \end{cases} \tag{2}$$

where $R', G', B' \in [0, 1]$ are gamma-corrected RGB primaries. YCbCr is dominantly used in compression applications since it reduces the redundancy in RGB color signals and represents the color with statistically independent components. The explicit separation of luminance and chrominance components and its wide adoption in image/video compression standards makes YCbCr popular for skin color analysis [7, 29, 33, 62].

3.1.3 HSV

The HSV coordinate system, originally proposed by Smith [57], defines color by: (i) Hue(H)—the property of a color related to the dominant wavelength in a mixed light wave, (ii) Saturation(S)—the amount of white light mixed with color that varies from gray through pastel to saturated colors, (iii) Value(V)—the property according to which an area appears to exhibit more or less light that varies from black to white. The HSV representation corresponds more closely to the human perception of color than aforementioned ones since it is derived from the intuitive appeal of the artist's tint, shade, and tone. The set of equations used to transform a point in the RGB to the HSV coordinate system ($[H, S, V] \in [0, 1]$, $[R', G', B'] \in [0, 1]$) is given as follows[4]:

$$\begin{cases} H &= \frac{1}{6} \times \begin{cases} 5 + (M - B')/C & \text{,if } M = R', m = G' \\ 1 - (M - G')/C & \text{,if } M = R', m \neq G' \\ 1 + (M - R')/C & \text{,if } M = G', m = B' \\ 3 - (M - B')/C & \text{,if } M = G', m \neq B' \\ 3 + (M - G')/C & \text{,if } M = B', m = R' \\ 5 - (M - R')/C & \text{,if } M = B', m \neq R' \end{cases} \\ S &= C/M \quad \text{(if C=0, then H = undefined)} \\ V &= M \end{cases} , \begin{cases} M = \max(R', G', B') \\ m = \min(R', G', B') \\ C = M - m \end{cases} \tag{3}$$

The HSV colorspace is traditionally shown as a hexcone model Fig. 3b and, in fact, HSV hexcone is a projection of the RGB cube along the gray-scale line. In hexcone model, H is represented as the angle and S corresponds to the horizontal distance from the vertical axis. V varies along the vertical axis with $V = 0$ being

[4] It is noteworthy to mention that the original literature [57] does not clearly indicate whether linear or nonlinear RGB is used in conversion. Although there is such an ambiguity, we use nonlinear RGB in this paper, which is implicit in image processing applications [52].

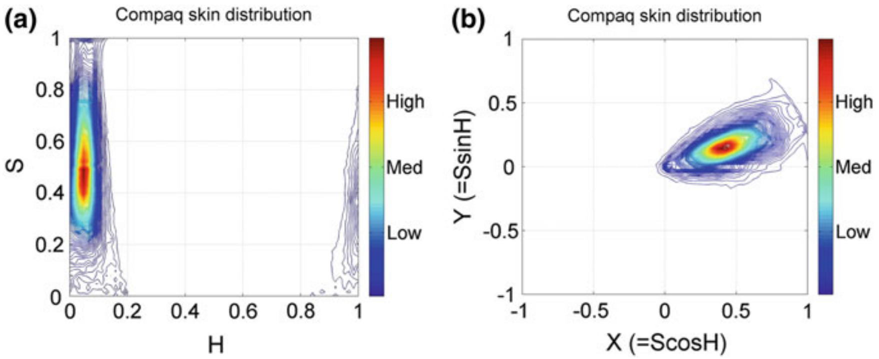

Fig. 4 Distribution of skin color pixel from Compaq dataset [36] in **a** H-S plane of original HSV colorspace, **b** Cartesian representation of H-S plane

black, and $V = 1$ being white. When $S = 0$, color is a gray value. When $S = 1$, color is on the boundary of hexcone. The greater the S, the farther the color is from white/gray/black. Adjusting the hue varies the color from red at $H = 0$, through green at $H = 1/3$, blue at $H = 2/3$, and back to red at $H = 0$. When $S = 0$, or $V = 0$, (along the achromatic axis) the color is grayscale and H is undefined.

Two chromatic components, H and S, are known to be less variant to changes of illumination direction and illumination intensity [20], and thus HSV is a reliable colorspace for skin color detection. However, one should be careful in exploiting HSV space for skin color analysis since manipulation of H involves circular statistics as H is an angular measure [27]. As can be seen in Fig. 4a, the cyclic nature of H component disallows use of a color distribution model which requires a compact cluster, e.g. a single Gaussian model, since it generates two separate clusters on both sides of H axis. To address this issue, polar coordinate system of H-S space can be represented in Cartesian coordinates X-Y as follows [4]:

$$X = S\cos(2\pi H), \quad Y = S\sin(2\pi H) \tag{4}$$

where $H, S \in [0, 1]$ and $X, Y \in [-1, 1]$. Here, X component can be regarded as a horizontal projection of color vector representing a pixel in H-S space, while Y component can be regarded as a vertical projection of color vector representing a pixel in H-S space. In the Cartesian representation of H-S space (Fig. 4b), skin color distribution forms a tightly distributed cluster.

3.2 Skin Color Distribution Model

Skin color distribution model defines a decision rule to discriminate between skin and non-skin pixels in given colorspace. To detect skin color pixel, a multitude

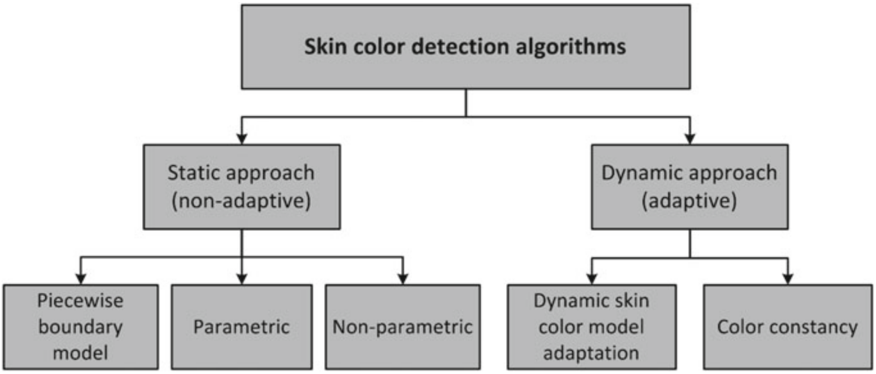
Skin color detection algorithms
Static approach (non-adaptive)
Dynamic approach (adaptive)
Piecewise boundary model
Parametric
Non-parametric
Dynamic skin color model adaptation
Color constancy

Fig. 5 Classification of skin color detection approaches

of solutions merge from two distinct categories (Fig. 5). The first type of methodologies use a static color classification rule and can be further divided into three sub-classes depending on the classifier type: (i) piecewise boundary model [7, 23, 30, 42], (ii) non-parametric skin distribution model [4, 36, 77], (iii) parametric skin distribution model [45, 73, 74]. However, static skin color classification methods are typically very sensitive to imaging conditions, e.g. skin color of same individual varies depending on the color temperature of light source (e.g. incandescent, fluorescent, and sunlight) as well as the characteristics of image acquisition devices (e.g. sensor spectral sensitivity and embedded white balancing algorithm). Therefore, it requires appropriate adaptation schemes to maintain stable performance in real-world environments. The dynamic approaches address such problematic cases either by pre-processing given input image to alleviate the influence of imaging condition on color description or by dynamically updating color classification model according to imaging condition.

Since skin color typically forms a small cluster in the colorspace, one of the easiest methods to build a skin color classifier is to explicitly define fixed decision boundaries of skin regions. Single or multiple ranges of threshold values for each color component are defined and the image pixel values that fall within these pre-defined ranges are defined as skin pixels. Piecewise boundary model has been exploited in various colorspaces and Table 1 presents some representative examples.

Non-parametric skin color modeling methods estimate the probability of a color value to be a skin by defining a model that has no dependency on a parameter. The most representative methods in this class is Jones and Rehg's method [36] which uses a 2D or 3D color histogram to represent the distribution of skin color in colorspace. Under this approach, the given colorspace is quantized into a number of histogram bins and each histogram bin stores the likelihood that a given color belongs to the skin. Jones and Rehg built two 3D RGB histogram models for skin and non-skin from Compaq database [36] which contains around 12K web images. Given skin and non-skin histograms, the probability that a given color belongs to skin and non-skin class is defined as:

Table 1 Commonly used piecewise boundary models for skin color detection

Authors (Colorspace)	Skin color classification rule
Kovac et al. [42] **(RGB)**	Uniform daylight illumination $R > 95, G > 40, B > 20, Max(R, G, B) - Min(R, G, B) < 15, \|R - G\| > 15, R > G, R > B$ Flashlight or daylight lateral illumination $R > 220, G > 210, B > 170, \|R - G\| \leq 15, B < R, B < G$ [a]
Gomez&Morales [23] **(nRGB)**	$\frac{r}{g} > 1.185, \quad \frac{rb}{(r+g+b)^2} > 0.107, \quad \frac{rg}{(r+g+b)^2} > 0.112$ [b]
Chai&Ngan [7] **(CbCr)**	$77 \leq Cb \leq 127, 133 \leq Cr \leq 173$ [c]
Herodotou et al. [30] **(HSV)**	$0.94 \leq H \leq 1$ or $0 \leq H \leq 0.14, 0.2 \leq S, 0.35 \leq V$ [d]

$$p(\mathbf{c}|w_s) = \frac{s(\mathbf{c})}{\text{Total skin pixel count}} \quad , \quad p(\mathbf{c}|w_n) = \frac{n(\mathbf{c})}{\text{Total non-skin pixel count}} \tag{5}$$

where $s(\mathbf{c})$ is the pixel count in the color bin $\mathbf{c}$ of the skin histogram, $n(\mathbf{c})$ is the pixel count in the color bin $\mathbf{c}$ of the non-skin histogram. For skin pixel detection, we need to estimate $p(w_s|\mathbf{c})$—a probability of observing skin pixel given a $\mathbf{c}$ color vector. To compute this probability, the Bayesian rule is applied using the given conditional probabilities of skin and non-skin:

$$p(w_s|\mathbf{c}) = \frac{p(\mathbf{c}|w_s)p(w_s)}{p(\mathbf{c}|w_s)p(w_s) + p(\mathbf{c}|w_n)p(w_n)} \tag{6}$$

Instead of calculating the exact value of $p(w_s|\mathbf{c})$, the ratio between $p(w_s|\mathbf{c})$ and $p(w_n|\mathbf{c})$ can be compared (i.e. likelihood ratio test) for classification as follows:

$$\begin{aligned} \mathbf{c} \in w_s \text{ ,if } & \frac{p(w_s|\mathbf{c})}{p(w_n|\mathbf{c})} > K \\ \Rightarrow \text{ if } & \frac{p(w_s|\mathbf{c})}{p(w_n|\mathbf{c})} = \frac{p(\mathbf{c}|w_s)p(w_s)}{p(\mathbf{c}|w_n)p(w_n)} > K \\ \Rightarrow \text{ if } & \frac{p(\mathbf{c}|w_s)}{p(\mathbf{c}|w_n)} > \theta \text{ ,where } \theta = K \times \frac{p(w_n)}{p(w_s)} \end{aligned} \tag{7}$$

where θ is an adjustable threshold that controls the trade-off between true positive (TP) and false positive (FP) rates (See 13 for definitions of TP and FP).

The third categories in static approaches are parametric skin color modeling methods where color classification rule is derived from parameterized distributions. The parametric models have the advantage over the non-parametric ones that they require smaller amount of training data and storage space. Key problems for parametric skin color modeling are to find the best model and to estimate its parameters. The most popular solutions include single Gaussian model (SGM) [73], Gaussian mixture model (GMM) [24, 32, 74], and elliptical model [45].

Under controlled environment, skin colors of different subject cluster in a small region in the colorspace and hence, the distribution can be represented by SGM [73]. A multivariate Gaussian distribution of a d-dimensional color vector $\mathbf{c}$ is defined as:

$$G(\mathbf{c}; \boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left[-\frac{(\mathbf{c}-\boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{c}-\boldsymbol{\mu})}{2}\right] \tag{8}$$

where $\boldsymbol{\mu}$ is the mean vector and Σ is the covariance matrix of the normally distributed color vector $\mathbf{c}$. The model parameters are estimated from the training data using the following equations:

$$\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{c}_i,\ \Sigma = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{c}_i - \boldsymbol{\mu})(\mathbf{c}_i - \boldsymbol{\mu})^T \tag{9}$$

Either the $G(\mathbf{c}; \boldsymbol{\mu}, \Sigma)$ probability or the Mahalanobis distance from the $\mathbf{c}$ color vector to the mean vector $\boldsymbol{\mu}$, given the covariance matrix Σ, can be used to measure the similarity of the pixel with the skin color.

Although SGM has been a successful model to represent skin color distribution, the assumption of SGM requires a single cluster which smoothly varies around the mean. However, such an assumption often causes intolerable error in skin/non-skin discrimination since different modes (due to skin color types and varying illumination conditions) can co-exist within the skin cluster. Therefore, Yang and Ahuja introduced the GMM model [74] to represent more complex shaped distribution. The GMM probability density function (pdf) can be defined as a weighted sum of Gaussians:

$$p(\mathbf{c}; \alpha_i, \boldsymbol{\mu}_i, \Sigma_i) = \sum_{i=1}^{N} \alpha_i G_i(\mathbf{c}; \boldsymbol{\mu}_i, \Sigma_i) \tag{10}$$

where N is the number of mixture components, α_i is the weight of i-th component ($\alpha_i > 0$, $\sum_{i=1}^{N} \alpha_i = 1$), G_i is a Gaussian pdf with parameters $\boldsymbol{\mu}_i$ and Σ_i. The parameters of a GMM are approximated from the training data via the iterative expectation-maximization (EM) technique [11].

Lee and Yoo [45] claimed that SGM is not accurate enough to approximate the skin color distribution because of the asymmetry of the skin cluster with respect to its density peak. They proposed an elliptical boundary model based on their observations that the skin cluster is approximately elliptic in shape. The elliptical boundary model is defined as:

$$\Phi(\mathbf{c}) = (\mathbf{c} - \boldsymbol{\phi})^T \Lambda^{-1} (\mathbf{c} - \boldsymbol{\phi}) \tag{11}$$

$\boldsymbol{\phi}$ and Λ are two model parameters to be estimated from training data:

$$\boldsymbol{\phi} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{c}_i \quad , \quad \Lambda = \frac{1}{N}\sum_{i=1}^{n} f_i(\mathbf{c}_i - \boldsymbol{\mu})(\mathbf{c}_i - \boldsymbol{\mu})^T \quad , \quad \boldsymbol{\mu} = \frac{1}{N}\sum_{i=1}^{n} nf_i\mathbf{c}_i \qquad (12)$$

where n is the number of distinctive training color vectors $\mathbf{c}_i$ of the training skin pixels, f_i is the number of skin samples of color vector $\mathbf{c}_i$, and N is the total number of samples ($N = \sum_{i=1}^{n} nf_i$). An input pixel $\mathbf{c}$ is classified as skin if $\Phi(\mathbf{c}) < \theta$ where θ is a threshold value.

3.3 Comparison and Discussion of Skin Color Distribution Models and Color Representations

Comparative assessment of skin color detection methods in different color representations has been discussed in many literatures [8, 41, 51, 53, 60, 63, 64], but they report different results mainly due to their different experimental conditions (e.g. selection of training/testing datasets). In this section, we will highlight some general conclusions derived from existing researches. Typically, the performance of skin color detection is measured using true positive rate (TPR) and false positive rate (FPR), defined as follows:

$$\begin{aligned} TP &= \frac{\text{number of correctly identified skin pixels}}{\text{total number of skin pixels}} \\ FP &= \frac{\text{number of falsely identified non-skin pixels as skin pixels}}{\text{total number of non-skin pixels}} \end{aligned} \qquad (13)$$

Most classification methods have an adjustable threshold parameter that controls the classifier decision boundary. As a result, each threshold value produces a pair of FP and TP values, generating a receiver operating characteristics (ROC) curve which demonstrates the relationship between TP and FP in different threshold values. For comparative evaluation of different classifiers, ROC performance is often summarized by a single scalar value, the area under ROC curve (AUC). AUC is known to be a fairly reliable performance measure of the classifier [14]. Since the AUC is a subregion of the unit square, its value lies between 0 and 1, and larger AUC value implies better classification performance.

A piecewise boundary model has fixed decision boundary parameters and hence, the corresponding ROC plots have only one point. Its boundary parameter values differ from one colorspace to another and one illumination to another. Although methods in this category are computationally fast, in general, they suffer from high FP rates. For example, Phung et al. [51] indicated that piecewise boundary classifier in CbCr space [7] achieves 92 % TP at 28 FP on Edith Cowan University (ECU) dataset [51] (consists of 4K color images from web or taken with digital camera, containing skin pixels), while both 3D SGM classifier of skin/non-skin (YCbCr) and Bayesian classifier with 3D histogram (RGB) achieves higher than 95 % TP at the same FP.

The Gaussian distribution based classifiers, e.g. SGM and GMM, and an elliptical model classifier have been widely used for skin color analysis since they generalize well with small amount of training data. In order to compare the performance of SGM and GMM, Caetano et al. [6] conducted comparative evaluation in normalized-rg colorspace using a dataset of 800 images from various ethnic groups (publicly not available). The authors noted that: i) GMM generally outperforms SGM for FP rates higher than 10 %, while both models yield similar performance for low FP rates, ii) detection performance remains unchanged except minor fluctuations when increasing mixture components for GMM from 2 to 8. Fu et al. [18] performed similar comparative assessment of both Gaussian classifiers using Compaq dataset [36] and confirmed that increasing mixture components doesn't provide significant performance improvement for $n > 5$ in four representative colorspaces (RGB, YCbCr, HSV, and normalized RGB). This is due to an overfitting issue, implying that a classifier describes training sample well but is not flexible enough to describe general samples. Moreover, using GMM is slower during classification since multiple Gaussian components must be computed to obtain the probability of a single color value. Therefore, one should be careful in selecting appropriate number of mixture components.

The performance of the representative non-parametric method, Bayesian classifier with 3D histogram [36], has been compared with other parametric approaches in [36, 51]. The histogram technique in 3D RGB color space achieved 90 % TP (14.2 FP) on Compaq database, slightly outperforming GMM or SGM in terms of detection accuracy. But it requires a very large training dataset to get a good classification rate, as well as higher storage space. For example, a 3D RGB histogram with 256 bins per channel requires more than 16 millions entries. To address this issue, some literature presented color bin quantization method to reduce color cube size. Jones and Rehg [36] compared the use of different numbers of histogram bins (256^3, 32^3, 16^3) and found that 32^3 histogram performed best, particularly when small amount of training data was used.

Often, skin detection methods solely based on color cue in Sect. 3.2 (summarized in Table 2) are not sufficient for distinguishing between skin regions and skin-colored background regions. In order to minimize false acceptance of skin-colored background objects as skin regions, textural and spatial properties of skin pixels can be exploited [10, 40, 69]. Such methods generally rely on the facts that skin texture is smoother than other skin similar areas. For example, Wang et al. [69] initially generated a skin map via pixel-wise color analysis, then carried out texture analysis using Gray-Level Co-occurrence Matrices (GLCM) features to refine the original skin map (i.e. remove false positives). Khan et al. [40] proposed a systematic approach employing spatial context in conjunction with color cue for robust skin pixel detection. At first, a foreground histogram of probable skin colors and a background histogram of the non-skin colors are generated using skin pixel samples in the input image extracted via a face detector. These histograms are used to compute foreground/background weights per pixel, representing the probability of each pixel being skin or non-skin. Subsequently, spatial context is taken into account by

Table 2 Summary of various skin color detection methods (In characteristic column, + and − represents pros and cons respectively)

Category	Method	Characteristic
Piecewise boundary	Chai (YCbCr) [7], Gomez (normalized RGB) [23], Herodotou (HSV) [30], Kovac (RGB) [42]	+ Simple implementation − Limited flexibility due to fixed threshold and high false positive rate
Non-parametric	Brown's self organizing map (SOM) [4], Jones's Bayesian approach with 3D histogram [36], Zarit's lookup table (LUT) [77]	+ Higher detection accuracy and less dependency on choice of colorspace − Require larger amount of training data and storage compared to parametric solutions
Parametric	Single Gaussian Model [73], Gaussian Mixture Model [24, 32, 74], Lee's elliptical model [45]	+ Better generalization with less training data − Potential long training delay (for mixture model) and high dependency on choice of colorspace

applying the graph-cut based segmentation on basis of computed weights, producing segmented skin regions of reduced false positives.

Selection of the best colorspace for skin classification is a very challenging task. This problem has been analyzed in numerous literatures with various combinations of skin color distribution models and training/testing datasets. In general, effectiveness of specific color representation in skin color detection can be measured based on their separability between skin and non-skin pixels, and robustness towards illumination variation. However, there is no single best colorspace that is clearly superior to others in all images and often only marginal improvement can be achieved by choice of colorspace.

The effectiveness of colorspace is also dependent on selection of skin color distribution model. For example, non-parametric models, such as histogram-based Bayes classifier are less sensitive to colorspace selection compared to parametric modeling schemes, such as SGM and GMM [1, 38, 65]. Some literatures indicate that transforming 3D colorspace to 2D by discarding the luminance component may enhance skin detection performance since chrominance components are more important cue for determination of skin color. However, elimination of luminance component should be avoided since it decreases classification performance [1, 38, 53], and therefore, is not recommended unless one wants to have faster solution (due to dimensionality reduction) at the cost of classification accuracy.

3.4 Illumination Adaptation for Skin Color Detection

Most of the skin color detection methodologies presented in Sect. 3.2 remain stable only to slight variation in illumination since the appearance of color is heavily dependent on the illumination condition under which the object is viewed. In order to maintain reliable performance over a wide range of illumination conditions, several illumination adaptation schemes have been proposed, which can be subdivided into two main approaches [38]: (i) Dynamic model adaptation: by updating trained skin color models dynamically according to the illumination and imaging conditions, (ii) Color constancy: by pre-processing an input image to produce a transformed version of the input as if the scene is rendered under standard illumination condition.

3.4.1 Dynamic Adaptation of Skin Color Model

Dynamic model adaptation approaches usually depend on the results of high-level vision tasks such as face detection and tracking to improve skin color detection performance. Sigal et al. [55] introduced a video-based dynamic adaptation scheme using a second-order Markov model to predict the transition of skin color model over time. Initially, they trained a non-parametric skin color classifier with RGB histogram using Compaq database in offline. This pre-trained model is used to segment the skin color region of the first frame of the input image sequence for tracking purpose. Then skin mask from the first frame is used to re-estimate histogram for skin and non-skin for subsequent frames. In particular, the time varying pattern of skin color distribution in colorspace is parameterized by affine transformations: translation, rotation, and scaling. From the comparative analysis against Jones and Rehg's static histogram approach [36], the authors reported enhanced skin segmentation accuracy in 17 out of 21 testing video sequences (containing illumination conditions ranging from white to non-white light sources, and shadows).

It is possible to adapt the skin color model even for single images, given that input image contains human face regions and they can be accurately located [3, 17, 39, 40, 78]. Zeng and Luo [78] presented an image-dependent skin color detection framework, where an elliptical skin color model trained offline in CIELAB colorspace, is adapted for the input image using a face detector. Skin pixels of given input image is extracted from the face region located by a Viola-Jones face detector [67] and a pre-trained elliptical skin color model is shifted towards the mean value of collected skin samples. During the model shift, the detection boundary threshold is tightened in order to effectively reduce false positives. Kawulok [39] also proposed a systematic solution, which dynamically updates a non-parametric skin color model by exploiting skin samples obtained from Support Vector Machine (SVM) based face detection module. The author noted that by applying the face detector in conjunction with Jones and Rehg's non-parametric method [36], the detection error rate (i.e. sum of false positive rate and false negative rate) is decreased from 26 % to 15 on ECU dataset [51] compared to the absence of the face detection operation. Aforementioned adaptation schemes taking advantage of skin samples extracted from face detectors are not only effective on varying illumination conditions but also beneficial on dealing with the skin color difference between individuals under constant illumination. Pre-trained color model tends to be more general than one estimated on basis of a present face, thereby fine-tuning the pre-trained model matching to an individual presented in the given image leads to reduction of false skin detections.

Sun [61] proposed an adaptive scheme without deploying additional high-level analysis. Initially, a global non-parametric skin color model [36] is trained in RGB colorspace and potential skin pixels are extracted from the input image via the global model. Then, pixels which are very likely to be skin are identified from initially extracted skin pixels using an accumulated histogram of skin likelihood ratio. For identified skin samples, a local GMM color model is constructed using K-mean clustering. Finally the globally trained skin model and the local skin model are linearly combined to produce a final adaptive model. The performance of this adaptive scheme depends on the number of selected skin pixel samples and the weighting factors for combining two models. The author indicated that it outperforms Jones and Rehg's method [36] on Compaq dataset in terms of detection accuracy, especially in the range of low false positive rate. This approach can be considered as a cost-effective alternative to aforementioned face detection based solutions, but it may not as accurate as them since it only makes use of color feature during adaptation.

Overall, the effectiveness of the dynamic skin color model adaptation depends on the validity of assumptions behind the adaptation criteria. The adaptation schemes generally use a general skin model obtained from a representative image set and then fine-tune it into an image specific model.

3.4.2 Color Constancy

Color constancy method attempts to minimize the effect of illumination and imaging condition by preprocessing the input image, instead of adjusting skin color classification model. Color constancy is the ability of human visual system HVS to recognize object color regardless of the illumination conditions. The aim of computational color constancy algorithm is to compensate the effect of the scene illuminant (i.e. light source) on the recorded image in order to recover underlying color of object [22]. Typically, color constancy algorithms can be viewed as a two-stage operation where the first step estimates the characteristics of scene illuminant from the image data, followed by the second step that applies correction on the image to generate a new image of the scene as if it were taken under a canonical (or reference) illuminant[5] (Fig. 6).

General formulation of color constancy

Let's consider an image acquisition device equipped with a lens that focuses light from a scene onto an array of sensors. If we assume the spectral power distribution (SPD) of the scene illuminant is constant, the illuminant can be specified by its SPD, $E(\lambda)$, which describes the energy per second at each wavelength $\lambda \in \mathbb{R}$. The light is reflected from surfaces of objects to be imaged and focused onto the sensor array. The

[5] For image reproduction applications, the canonical illuminant is often defined as an illuminant for which the camera sensor is balanced [2].

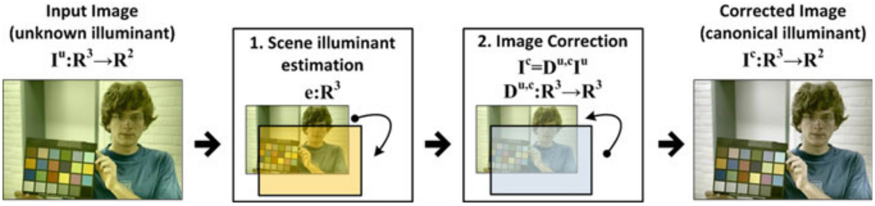

Fig. 6 Overview of color constancy procedure (Sample image is taken from [19])

property of light reflected from an object toward location $\mathbf{x}$ of the sensor is determined by the surface spectral reflectance, $S(\mathbf{x}, \lambda)$. Here, $\mathbf{x} \in \mathbb{Z}^2$ denotes the spatial position on the 2D sensor array at which the object is imaged. The light arriving at each location $\mathbf{x}$ on the sensor array is described by the function $E(\lambda)S(\mathbf{x}, \lambda)$.

Now, we assume that there are p-distinct classes of sensors at each location $\mathbf{x}$. Typical digital camera devices sample the light by Red, Green, and Blue sensors (i.e. $p = 3$). We denote the sensor spectral sensitivity of the k-th color channel as $\rho_k(\lambda)$ (($k \in \{R, G, B\}$)). According to Lambertian reflection model[6], these spectral functions are translated into the RGB values $\mathbf{I}(\mathbf{x}) = [I_R(\mathbf{x}), I_G(\mathbf{x}), I_B(\mathbf{x})]^T$ within the sensor as follows:

$$I_k(\mathbf{x}) = \int_\omega \rho_k(\lambda)E(\lambda)S(\mathbf{x}, \lambda)d\lambda \tag{14}$$

where the integral is taken over the entire visible spectrum ω (wavelength of approximately from 400 nm to 700). Eq. (14) shows that the responses induced in a camera sensor depend on the spectral characteristics of the illuminant and the surface. Essentially, color constancy problem can be posed as recovering an estimate of illuminant spectra $E(\lambda)$ or its projection on the RGB space

$$\mathbf{e} = [e_R, e_G, e_B]^T = \int_\omega \boldsymbol{\rho}(\lambda)E(\lambda)d\lambda \tag{15}$$

from given sensor responses $\mathbf{I}(\mathbf{x})$.

Without prior knowledge, the estimation of $\mathbf{e}$ is an under-constrained problem. In practice, color constancy algorithms rely on various assumptions on statistical properties of the illuminants, and surface reflectance properties to estimate $\mathbf{e}$. Once the illuminant is estimated, then all colors in the input image, taken under an unknown illuminant, are transformed to colors as they appear under the canonical illuminant. Each pixel of the image under an unknown illuminant $\mathbf{I}^u = [I^u_R, I^u_G, I^u_B]^T$ can be mapped to the corresponding pixel of the image under a canonical illuminant $\mathbf{I}^c =$

[6] Lambertian reflection model explains the relationship between the surface reflectance and color image formation for flat, matte surfaces. Although this model does not hold true for all materials, it provides a good approximation in general, and thus widely used in design of tractable color constancy solutions

$[I^c_R, I^c_G, I^c_B]^T$ by a transformation matrix, $\mathcal{D}^{u,c} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$

$$\mathbf{I}^c = \mathcal{D}^{u,c}\mathbf{I}^u \tag{16}$$

Most existing algorithms make use of a diagonal matrix for this transformation. The diagonal model maps the image taken under an unknown illuminant to another simply by treating each channel independently:

$$\mathbf{I}^c = \mathcal{D}^{u,c}\mathbf{I}^u \quad \Rightarrow \quad \begin{pmatrix} I^c_R \\ I^c_G \\ I^c_B \end{pmatrix} = \begin{pmatrix} d_R & 0 & 0 \\ 0 & d_B & 0 \\ 0 & 0 & d_G \end{pmatrix} \begin{pmatrix} I^u_R \\ I^u_G \\ I^u_B \end{pmatrix} \tag{17}$$

where diagonal entries of $\mathcal{D}^{u,c}$ can be computed as:

$$d_k = e_k \Big/ \left\{ \sqrt{3(e^2_R + e^2_G + e^2_B)} \right\} \tag{18}$$

This model is derived from the Von Kries hypothesis [43] that human color constancy is an independent gain regulation of the three cone signals, through three different gain coefficients.

Representative color constancy algorithms

Various color constancy algorithms have been proposed in the literatures [22], and they can be categorized into two main groups: (i) static approach : estimates the illuminant solely based on the content of a single image with certain assumptions on general nature of color images, (ii) learning approach : requires training data in order to build a statistical model prior to estimation of the scene illuminant. Among available solutions, static approaches, such as Grayworld [5], White Patch [44], Shades of Gray [15], and Gray Edge [71] have been widely used in practical applications due to their simple implementation and fast execution speed. The most widely used Grayworld (GW) algorithm [5] assumes that the average reflectance in a scene is gray (i.e. achromatic) under a neutral illuminant, and thus any deviation from gray is caused by the effects of the illuminant. Hence, the RGB value of the illuminant in the image $\mathbf{I}$, $\mathbf{e} = [e_R, e_G, e_B]^T$, can be estimated by computing the average pixel value:

$$\int_{\mathbf{x}} \mathbf{I}(\mathbf{x})d\mathbf{x} = k\mathbf{e} \tag{19}$$

where $\mathbf{I}(\mathbf{x})$ is RGB value of the two-dimensional spatial coordinates $\mathbf{x} \in \mathbb{Z}^2$, and k is a multiplicative constant. Another popular algorithm, the White Patch (WP) [44] assumes that a surface with perfect reflectance property[7] exists in the scene, and the color of the perfect reflectance is the color of the scene illuminant:

[7] A surface with perfect reflectance property reflects the incoming light in the entire visible spectral range (between wavelengths of about 400 and 700 nm of the electromagnetic spectrum)

$$\max_{\mathbf{x}} \mathbf{I}(\mathbf{x}) = \left[\max_{\mathbf{x}} I_R(\mathbf{x}), \max_{\mathbf{x}} I_G(\mathbf{x}), \max_{\mathbf{x}} I_B(\mathbf{x})\right]^T = k\mathbf{e} \tag{20}$$

Finlayson and Trezzi [15] demonstrated that GW and WP are two different instantiations of a more general color constancy algorithm based on the Minkowski norm. This method is called Shades of Gray (SoG) and is computed by:

$$\left[\int (\mathbf{I}(\mathbf{x}))^p d\mathbf{x}\right]^{(1/p)} = k\mathbf{e} \tag{21}$$

where p is the Minkowski norm. For $p = 1$, the equation is equivalent to the GW assumption, while for $p = \infty$, it is equivalent to color constancy by WP. The authors investigated the performance of the illuminant estimation with various p values and reported that the best results are obtained with a Minkowski norm of $p = 6$ across many dataset.

Aforementioned methods (GW, WP, and SoG) use only RGB pixel values to estimate the illuminant of an image, completely ignoring other information. More recently, Weijer et al. [71] extended pixel based color constancy methods to incorporate derivative information, which resulted in the Gray Edge (GE) algorithm. Gray edge is based on the hypothesis that the average of the reflectance differences in a scene is achromatic. Under Gray Edge assumption, the color of light source can be computed from the average color derivative in the image given by:

$$\left[\int |\mathbf{I}^{\sigma}_{\mathbf{x}}(\mathbf{x})|^p d\mathbf{x}\right]^{(1/p)} = k\mathbf{e} \tag{22}$$

where subscript $\mathbf{x}$ indicates the spatial derivative, and $\mathbf{I}^{\sigma}$ is a convolution of the image $\mathbf{I}$ with a Gaussian smoothing filter $\mathbf{G}$ with standard deviation σ.

It is worthwhile to note that due to under-constrained nature of illumination estimation problem, no single color constancy method is superior to others in all images and it may yield suboptimal result when its underlying assumption doesn't hold true. Aforementioned methods assume that the scene is lit by a single illuminant; although in reality this assumption is often violated due to the presence of multiple illuminants. Furthermore, assumptions on scene statistics may not be correct for given input image. For example, the underlying assumption of GW algorithm fails when the image contains dominant colors in the scene (e.g. image of forest, or ocean), since the average color won't be gray. In such cases, the application of GW algorithm will result in under or over-compensated scene. Such problems can be addressed by exploiting more advanced solution, such as the one based on extensive training data or complex assumptions. However, in general, aforementioned four color constancy algorithms are known to yield acceptable performance at low computational cost, and thus are suitable for practical face detection systems [12].

4 Case Study 1 : Use of Skin Color Cue in Image-Based Face Detection System

In this section, we demonstrate a case study of usage of color cue in the practical face detection system. A face detection framework exploiting a representative image-based approach is designed and a skin color classification module is integrated into the system to provide complementary information. In order to reduce the effect of scene illumination on detection performance, illumination compensation is performed on the input color image prior to facial analysis. The overview of the proposed framework is illustrated in Fig. 7.

For this study, we exploit the Boosting based face detection framework with LBP feature due to its superior performance in real-life face detection applications. Among LBP variants, MBLBP [80] is selected since: (i) MBLBP feature is an advanced version of the original LBP feature with high discriminative power, capable of extracting not only local texture as the original LBP, but also larger-scale structure information, (ii) Computation of MBLBP feature can be very fast by using integral image. Consequently, this particular case study allows us to demonstrate the contribution of skin color cue within state-of-the-art texture-based face detection framework. To build MBLBP based detector, we adopted the training procedure and design of Zhang et al.'s framework in [80] (More detail is provided in Sect. 4.1.3).

Following two experiments have been conducted for the performance evaluation.

1. The effectiveness of skin color cue in terms of improving detection accuracy and computational efficiency of the texture based facial analysis is evaluated by comparing two scenarios:

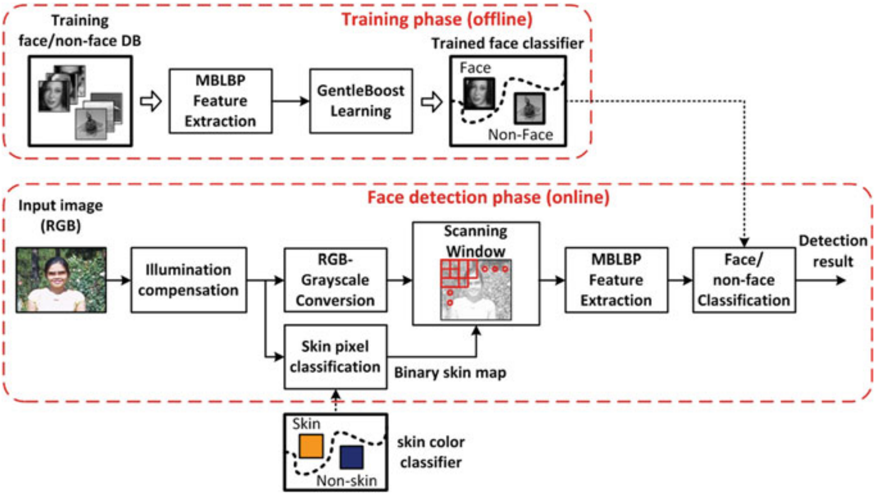

Fig. 7 Overview of the proposed image-based face detection pipeline exploiting MBLBP feature and skin color cue

- Texture feature only pipeline : Face detection is carried out without exploiting color information to measure the performance of the detector solely based on grayscale texture. To facilitate this scenario, illumination compensation and skin color classification blocks are disabled and all remaining operations are performed in grayscale domain.
- Hybrid pipeline (color feature in conjunction with texture) : Skin color classification block is enabled, and the binary skin map is generated from input color image to identify skin pixels.

2. The importance of stable color representation in face detection analysis is demonstrated. Several representative color constancy methods are applied to color input image to eliminate the color bias caused by non-standard illumination condition. Then, comparative assessment is done by comparing detection accuracy with illumination compensation enabled and disabled.

4.1 Proposed Face Detection Framework

The proposed face detection framework consists of three main components: (i) illumination compensation, (ii) skin color classification, (iii) MBLBP feature based face detection module. In this section, brief overview of each component is provided.

4.1.1 Illumination Compensation

Illumination compensation module allows the face detection framework to maintain stable performance over wide range of illumination conditions. In this module, the input RGB color images $X : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ is processed by color constancy algorithm to produce the corrected RGB image $I : \mathbb{R}^2 \rightarrow \mathbb{R}^3$. In this experiment, we make use of four representation color constancy algorithms: Grayworld, White Patch, Shades of Gray, and Gray Edge. They are cost-effective algorithms with simple implementation and thus, suitable for practical image processing applications.

4.1.2 Skin Color Classification

For the corrected color image $I : \mathbb{R}^2 \rightarrow \mathbb{R}^3$, skin color classification module performs a pixel-wise binary classification and generates a binary skin map, $sMap : \mathbb{R}^2 \rightarrow \mathbb{R}$, where $sMap(\mathbf{x}) = 1$ if $I(\mathbf{x}) \in w_s$, while $sMap(\mathbf{x}) = 0$ if $I(\mathbf{x}) \in w_n$ ($\mathbf{x} \in \mathbb{Z}^2$ is two-dimensional spatial coordinates in the image). In order to perform classification (online), statistical color models are constructed using training skin/non-skin samples (offline). In our experiment, GMM models are derived to represent both skin and non-skin color distributions and the likelihood ratio test is applied to classify each pixel into skin/non-skin class. In other words, $p(\mathbf{c}|w_s)$

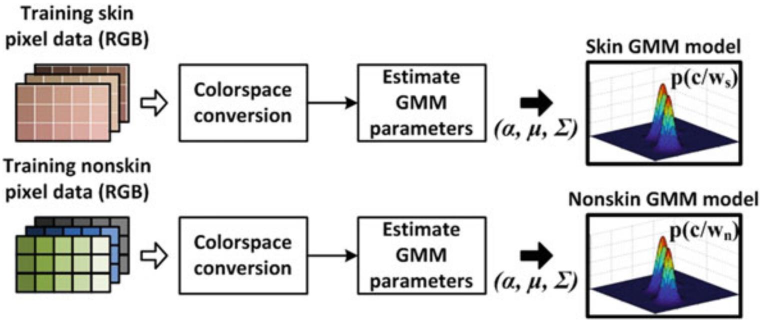

Fig. 8 Overview of skin color classifier training process

and $p(\mathbf{c}|w_n)$ are directly computed from given skin and non-skin Gaussian models, and $\frac{p(\mathbf{c}|w_s)}{p(\mathbf{c}|w_n)}$ is compared with a threshold value for classification as described in 7. GMM is selected due to following reasons: (i) it generalizes well with relatively small number of training samples, (ii) compared to other parametric models, it provides more reliable means to represent multi-modal distribution of human skin color under varying illumination.

Skin Classifier Training/Testing

We examined a GMM based Bayesian skin classifier with various combinations of mixture components and colorspaces to identify its optimal configuration. Skin color classifiers are trained in five commonly used colorspaces in skin color analysis, including RGB, normalized RGB, YCbCr, HSV, and Cartesian-HSV (denoted as cHSV). For each colorspace, we examined up to four Gaussian mixtures for skin/non-skin pair, allowing us to test 20 combinations in total[8]. In order to train GMM models of skin and non-skin color distributions, a dataset of 1923 images (1014 containing human skin pixels, 909 without human skin pixels) are collected from world wide web. The skin subset contains images of Asian (311 images), Caucasian (319), and Dark skin group (384), while the non-skin subset contains images of art/illustrations (233), Natural Scene (558), and Product (118). All collected images are in JPEG format with sRGB 8-bit representation, and uniform in spatial resolution as 1024 × 768 (both landscape/portrait format) to ensure each image has almost equivalent contribution in total training pool. For training, skin and non-skin pixels in training data are converted to each colorspace and GMM parameters are estimated by EM algorithm along with k-mean clustering initialization, adopting a recommendation from [74] (Fig. 8).

The skin pixel classification performance was tested on 2000 color images from Compaq database [36], containing skin color pixels from various ethnic backgrounds

[8] Assigning more mixture components for non-skin class than skin class is beneficial due to less compact shape of non-skin sample distribution. However, we found that performance gain from having more components for non-skin class is marginal and thus we maintain the same number of components for both classes in this experiment.

Table 3 The AUC index of skin color classification performance of 20 combinations

Number ofmixtures	Colorspace				
	RGB	HSV	YCbCr	cHSV	nRGB
1	0.8901	0.8723	0.8901	0.8868	0.8513
2	0.8967	0.8991	0.8928	0.8979	0.8772
3	0.8956	0.8932	0.8955	0.8970	0.8788
4	0.8961	0.8923	0.8967	0.8996	0.8713

and illumination conditions. This database is used for validation since it is one of the most frequently used benchmark databases from research community with predefined groudtruth information. Images in Compaq database are given in JPG or GIF file format with sRGB 8-bit representation. Detection performance of various experimental configurations are compared in AUC measure (Table 3).

Following observations are made from this experiment:

1. The normalized RGB provides the worst classification power among five colorspaces since luminance component is lost during its transformation from 3D to 2D, implying that discarding luminance component should be avoided to achieve highly accurate skin color classification.
2. GMM outperforms SGM in all five colorspaces, and particularly in HSV colorspaces where skin color distribution doesn't form a compact cluster. In general, increasing mixture components results in high classification performance upto certain number of components before overfitting occurs (e.g. for HSV colorspace, having more than two mixture components degrades detection performance). We observed that the best skin detection performance can be achieved by using Cartesian HSV colorspace with four mixture components for both skin and non-skin GMM models. Therefore, it is selected as our main trained classifier model and used throughout subsequent sections.

4.1.3 MBLBP Feature Based Face Detection System

This section briefly describes the MBLBP feature based detection system which contains all remaining modules in Fig. 7 except aforementioned two modules. Similarly to Viola-Jones's framework [67], the proposed framework performs exhaustive search on the input image to determine the existence of the face. This requires the grayscale version of input, which can be obtained by converting input image I from RGB to YCbCr and retaining only Y channel. Instead of scanning whole image for face search, the proposed system uses the generated binary skin mask $sMap$ to narrow down search region. Adopting the strategy from [12], particular sub-window is examined during face search, only if it contains sufficient number of skin color pixels. Let W_k is the kth sub-window examined during iterative window scan to determine whether this sub-window is face image or not. MBLBP based face/non-face classification is carried out only if:

$$\frac{\sum_{\mathbf{x}\in W_k} sMap(\mathbf{x})}{w_k \times h_k} \geq L \tag{23}$$

where w_k and h_k are the width and height of window W_k, respectively, and L is the threshold for minimum skin pixel count. For our testing dataset, we found $L = 0.4$ yields satisfactory results.

MBLBP based face detector training
To train MBLBP based face detector, 9916 gray scale face images in 24 × 24 pixels are collected by combining Viola and Jones dataset and Ole Jensen dataset [34]. These face images contain large variations including but not limited to pose, facial expression, and illumination conditions. The baseline size (24 × 24 pixels) is applied directly to training as in [67]. Furthermore, more than 100K of negative training image patches are extracted from 2,000 high resolution non-face images collected from the web. For consistency, the negative image patches are resized to the same size as face image patches. From the entire set, 7,916 face images and 10,000 non-face images are randomly extracted to train classifier, and another independent 2,000 face images and 10,000 non-face images are randomly selected for validation.

The face detector is trained aiming for high detection accuracy of approximately 97 % detection rate. To achieve this, the stage classifiers are trained to have higher than 99 % true positive rate (TPR) on validation dataset. In addition, a pre-assigned false positive rate (FPR) is achieved on validation dataset by adjusting the number of features and threshold for each stage. For fast processing, only a small number of features are used in the initial stage allowing a relatively high FPR (around 50 %). Each succeeding stage is designed to use increased number of features to reduce FPR by around 10 % from its prior stage. The number of stages is increased until the desired overall performance is achieved. Overall, the face detector achieves 96.9 % TPR and $7.94x10^{-6}$ FPR on validation dataset. This particular TPR and FPR configuration is chosen adopting general practice from the literatures[9]. GentleBoost machine learning algorithm is used to train the classifier due to its exceptional performance among the variants of Boosting algorithm [46].

4.2 *Experimental Results*

In order to evaluate the proposed face detection method, we have chosen the Bao color face database [16]. The Bao database is chosen due to following two reasons: (i) it represents real-life scenarios by containing wide variety of images, including faces of various ethnic groups (Asian, Caucasian, and Dark skin group), poses (frontal and non-frontal), illumination conditions (indoor and outdoor), and image

[9] Viola and Jones [67] indicate that around 1×10^{-6} of FPR is a common value for practical uses. However, it is extremely difficult to achieve the precise value and generally it is acceptable if FPR is within the same magnitude. For instance, Jun and Kim [37] achieves 96 % TPR at 2.56×10^{-6} FPR, and Louis and Plataniotis [47] achieves 92.27 % TPR at 6.2×10^{-6} FPR

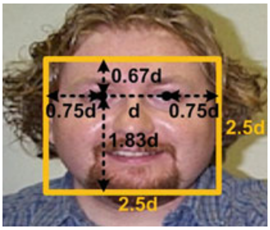

Fig. 9 Example of groundtruth face box generation (Sample image is taken from [16])

resolutions (from 57 × 85 to 1836 × 1190), (ii) it is publicly available database which is widely used for evaluation of face detector [12, 68]. All images in this dataset are provided in 8-bit JPEG format and we used 117 images containing a single face and 220 images containing multiple faces, examining a total of 1360 faces. Since the original Bao database does not contain the groundtruth information for the face locations, we manually generated groundtruth by locating a rectangle using the eye locations (Fig. 9). The correctness of face detection hypothesis is evaluated based on the groundtruth and the following two criterions [70]: (i) the Euclidean distance between the hypothesis face box center and groundtruth face box center must be within 30 % of the groundtruth width, (ii) the detection hypothesis width must be within 50 % of the groundtruth width.

In Fig. 10, the Free Receiver Operator Characteristic (FROC) is illustrated to compare two detection pipelines: MBLBP texture only pipeline (denoted as MBLBP) and hybrid of texture and color pipeline (denoted as MBLBP SS). FROC is similar to ROC except it plots the detection rate (DR) versus number of false positive detections instead of false positive rates. As can be seen, by incorporating color information, the proposed MBLBP feature based face detection yields enhanced detection accuracy over almost entire range of number of false positive (nFP). Since the texture only pipeline discards the chrominance information, it will generate false positive if the scene contains a background object of face-like texture pattern. By using skin color information, such false positives can be successfully filtered out during face search. The degree of improvement is approximately 1 % increase in TP from 93 % to 94 at nFP of 40. In Fig. 11, the detection results of both pipelines are compared on three samples images from Bao database, containing various skin types in indoor and outdoor lighting conditions. The figures demonstrate that all false positives have been successfully eliminated by exploiting skin color cue.

To compare computational complexity of both pipelines, we measure the total number of scanned sub-windows during face search for all images in Bao dataset. The number of scanned sub-windows is an important cue for computational speed, since only a subset of sub-windows which are sufficiently populated with skin color pixels are scanned for hybrid pipeline while all sub-windows have to be scanned for texture only pipeline. In addition, we measure the total execution time to process all 337 images in Bao dataset to evaluate computational speed in real MATLAB

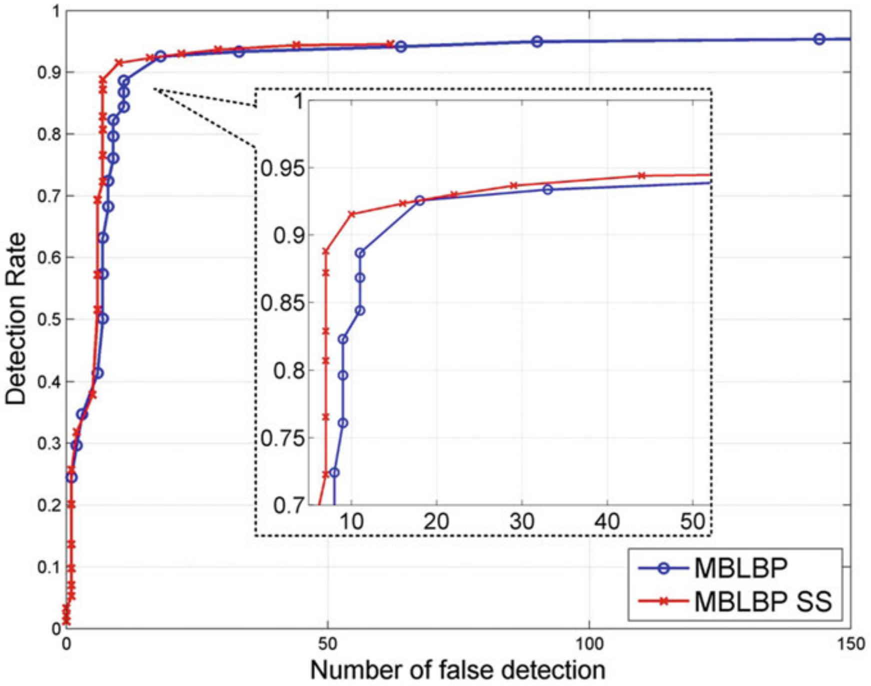

Fig. 10 FROC of face detection result on Bao dataset using MBLBP texture only pipeline and hybrid pipeline

Table 4 Computational complexity comparison between two pipelines

		MBLBP-texture	Hybrid
Total number of scanned sub-window during exhaustive search		43387629	9150270
Total processing delay	(in seconds)	62930	14286

implementation[10]. Experimental results are obtained on Core 2 Duo 3.0 GHz CPU with 4GB RAM running Windows 7 operating system. Although there is software overhead in MATLAB implementation, the obtained results are proportional to the number of scanned windows, as shown in Table 4. In hybrid pipeline, only 2.75 % of total execution delay (i.e. 393s out of 14286s) accounts for skin color detection, while more than 95 % of delay accounts for texture analysis, demonstrating computational efficiency of color analysis. Overall, hybrid pipeline not only allows us to enhance detection accuracy by removing false positives with non-skin color, but also significantly reduces computational complexity.

[10] Instead of measuring average number of scanned window and execution time per frame, we measured the sum of them, since test images in Bao database vary in spatial resolutions

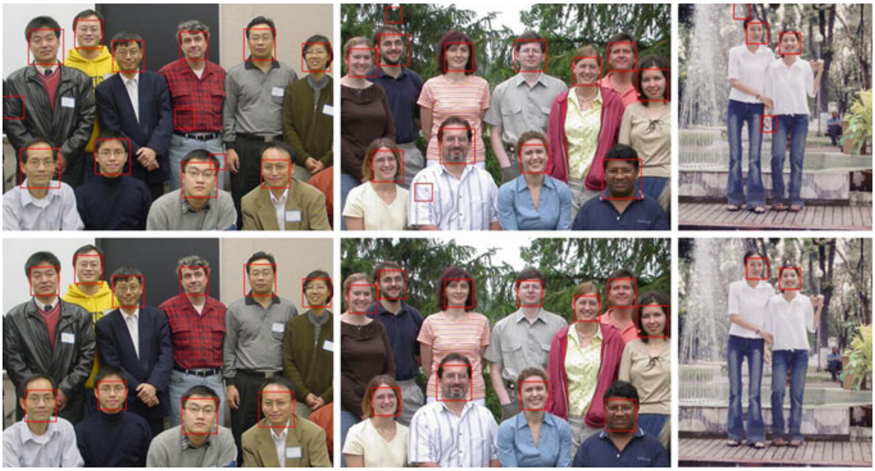

Fig. 11 Detection results of the proposed face detector for sample images from the Bao database [16]: upper images are obtained with MBLBP texture only pipeline whereas lower images are obtained with hybrid pipeline

Effectiveness of color constancy solution is evaluated in two different aspects that: (i) color constancy algorithm is applied to images under abnormal lighting condition and its impact on face detection performance is analyzed, (ii) color constancy algorithm is applied to images under generic lighting condition (e.g. images from standard image database) and its impact on face detection performance is analyzed. Such analysis allows us to identify color constancy solutions that improves detection performance on images under challenging illumination condition while providing comparable performance on general images.

Figure 12 presents a sample image rendered under indoor lighting with yellow cast, and adjusted images using four color constancy algorithms. The skin pixels of each image are manually extracted, and their RGB values are converted to HSV and plotted in the H-S plane. As can be seen, the skin color distribution of the original image is slightly biased towards yellow hue than the one under normal lighting condition (Refer to Table 1 for commonly accepted hue values for skin color). Consequently, skin color classifier fails to detect most of skin pixels in the scene, which eventually results in face localization error. By applying color constancy algorithm, the reliability of skin color classification is improved (e.g. with GW, SoG, and WP algorithms), allowing subsequent texture analysis to be performed in detected skin region. However, Fig. 12b shows that failure of illumination compensation may lead to even severe color distortion on skin pixels. It demonstrates that the performance of color constancy is image dependent and thus, to obtain meaningful measure of algorithm accuracy, average performance over a set of images should be assessed. In Fig. 13, the effectiveness of color constancy algorithms are evaluated on Bao database. Experimental results indicate that applying color constancy algorithms on large-scale image dataset generally yields comparable or slightly higher detection

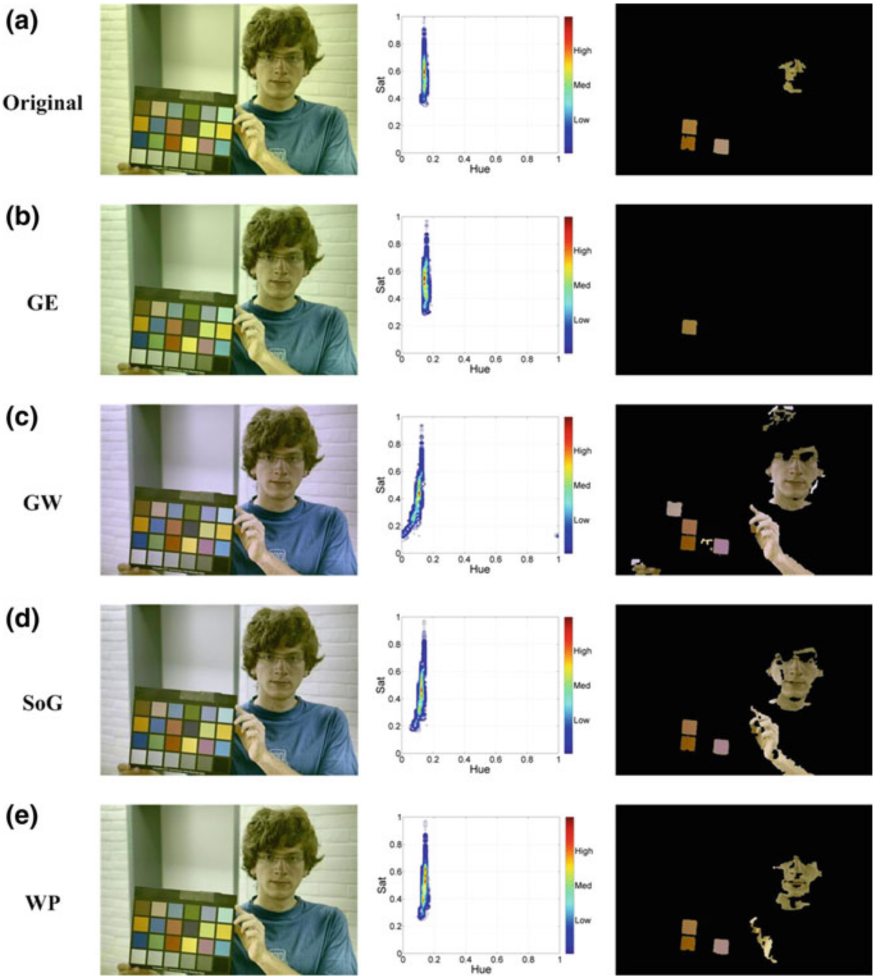

Fig. 12 Skin color pixel distribution in H-S plane and skin pixel detection results on a sample image taken from Color Checker Dataset [19] : **a** Original image, **b–e** images processed by GE, GW, SoG, and WP color constancy algorithms, respectively

performance, except for the GW algorithm, where DR drops significantly by approximately 5 %. It implies that redundant illumination compensation on images under generic illumination condition may deteriorate face detection performance and therefore algorithm should be carefully validated over a wide variety of images to build robust real-world solution.

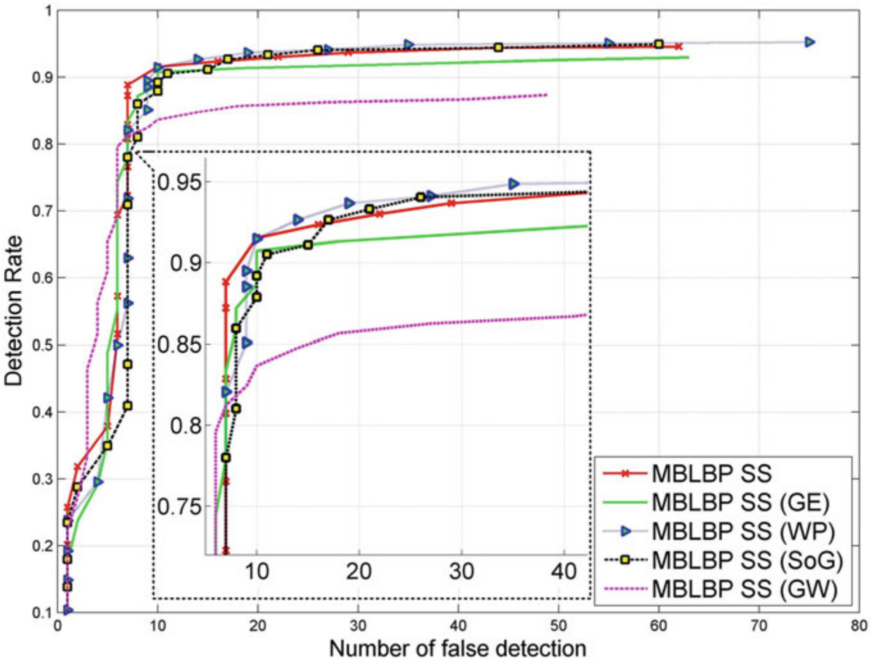

Fig. 13 FROC of face detection result on Bao dataset using hybrid pipeline with various illumination compensation methods

5 Case Study 2 : Use of Skin Color Cue in Feature-Based Face Detection System

In Sect. 4, we have demonstrated that color provides complementary information to reduce falsely detected faces, while minimizing face search delay in image-based face detection system. However, the aforementioned detection system has two major limitations: (i) even with complementary color information, it is still found to be computationally intensive due to exhaustive search process, (ii) its performance drops significantly when face is not frontal upright since training of system is done with frontal upright faces. Several advanced face detection methodologies have been proposed to achieve rotation invariant detection by cascading N pose specific detectors in parallel [66], but often such system exhibits increased complexity, fails to meet real-time requirements.

In this section, we demonstrate another example of face detection system, where aforementioned limitations of image-based approaches are addressed by using feature-based face detection approach in conjunction with color analysis. As mentioned earlier, color is an attribute that is invariant to rotation and scaling, and therefore, such property can be exploited to improve the robustness of face detection

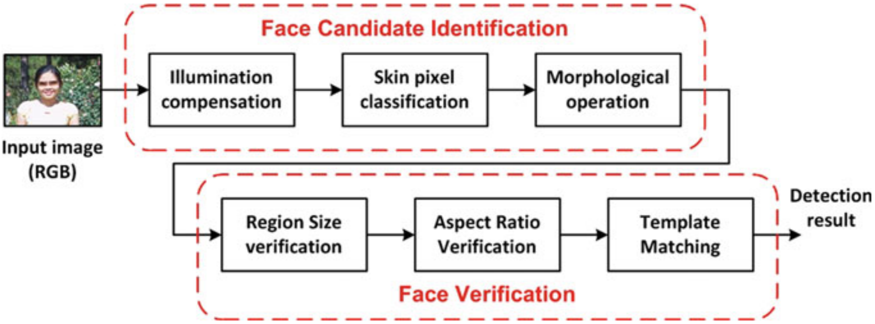

Fig. 14 Overview of feature-based face detection pipeline exploiting skin color cue

system towards rotated face. The workflow of the proposed face-detection algorithm presented in Fig. 14 can be summarized as follows:

- Illumination compensation is performed on the input color image to reduce the effect of prevailing illumination, followed by skin pixel classification to extract skin-like pixels
- By utilizing a series of binary morphological operations, face candidate regions are formed from detected skin pixels
- Each face candidate regions are verified through several analysis to determine whether if corresponding region is a face or not

5.1 Proposed Face Detection Framework

5.1.1 Face Candidate Identification

We start by extracting skin pixels using the GMM based classifier in Cartesian HSV colorspace outlined earlier in Sect. 4.1.2. Once all of the pixels have been classified by generating binary skin map, $sMap : \mathbb{R}^2 \rightarrow \mathbb{R}$, a series of binary morphological operations are subsequently applied to $sMap$ to refine the extracted skin region. Morphological operation simplifies image data by eliminating detail smaller than the structuring element while preserving their global structural information [28].

We utilize two morphological operations in series: morphological closing followed by morphological opening. Both operations are carried out with a disk-shaped structuring element of radius 3, which provide a good balance between noise reduction and structural detail preservation. Essentially, closing an image with a disk shape smoothes contour; eliminates small holes; and fills gaps within the objects, while opening an image with a disk shape eliminates small islands and sharp peaks. Consequently, object boundaries are smoothen out and small artifacts are effectively removed, achieving semantically meaningful segmentation. Subsequently, cluster of connected pixels are obtained from the refined binary image through connected component labelling to generate set of face candidates to be verified individually.

5.1.2 Face Verification

The input to face verification mechanism may contain objects other than the facial areas, such as other part of human body or skin-colored background objects. The verification module exploits general characteristics of human face region to distinguish actual face from a set of candidates. More specifically, we consider following criteria in sequential manner to verify face candidates:

1. Region Size : Any candidate regions of insignificant size are discarded from further verification by imposing minimum size constraint. Clusters of area less than 0.5 % of the image dimension are eliminated since such small regions generally don't correspond to actual face region if image was captured from reasonable distance.
2. Aspect Ratio : Given the geometry of the human face, it is reasonable to expect that the ratio of height to width falls within a specific range. If the dimensions of a candidate object satisfy the commonly accepted dimensions of the human face, then it can be classified as a facial area.

   To examine aspect ratio, each face candidate region F is approximated by the best-fit ellipse using statistical moments [58]. The ellipse is represented by a state parameter $\mathbf{s} = (x_F, y_F, \theta_F, a_F, b_F)$, where (x_F, y_F) is the center of ellipse, θ_F is the angle of major axis of the ellipse with the horizontal axis, a_F and b_F are the length of minor and major axis of the ellipse (Fig. 15). The parameters are obtained by finding an ellipse that has same normalized central moment as the candidate region. Initially, (x_F, y_F) is defined as the centroid of the region F. The normalized second central moments of the region F can be computed $(p + q = 2, p \geq 0, q \geq 0)$:

$$\overline{\mu}_{p,q}(F) = \mu_{p,q}(F) \cdot \left(1/\mu_{0,0}(F)\right)^{(p+q+2)/2} \tag{24}$$

   where $\mu_{p,q}(F)$ is the central moment of the region F, defined as:

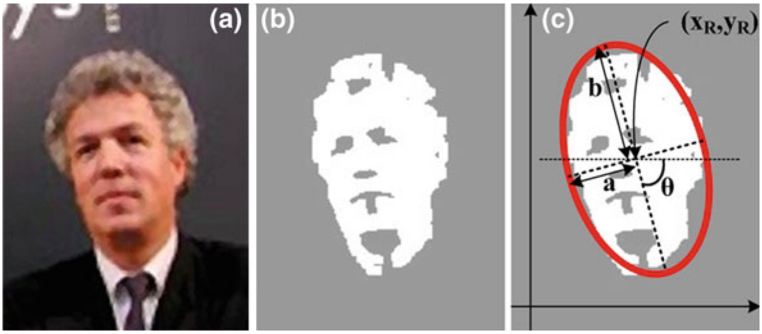

Fig. 15 Face candidate region represented by ellipse: **a** Original image, **b** Binary skin map of image **a**, **c** Ellipse matched to face candidate region

$$\mu_{p,q}(F) = \sum_{(i,j)\in F} (i - x_F)^p \cdot (j - y_F)^q \tag{25}$$

The orientation θ_F of the major axis ($\theta_F \in [-\pi/2, \pi/2]$) can be found from the central moments:

$$\theta_F = \frac{1}{2} \arctan\left(\frac{2 \cdot \mu_{1,1}(F)}{\mu_{2,0}(F) - \mu_{0,2}(F)}\right) \tag{26}$$

The length of minor and major axis, a_F and b_F can be computed as:

$$a_F = 2\sqrt{(\lambda_1/|F|)} \quad , \quad b_F = 2\sqrt{(\lambda_2/|F|)} \tag{27}$$

where $|F|$ is number of pixels in region F, and

$$\begin{gathered}\lambda_1 = \sqrt{(\mu_{2,0}(F) + \mu_{0,2}(F))/2 - \sqrt{4\mu_{1,1}(F)^2 + (\mu_{2,0}(F) - \mu_{0,2}(F))^2}} \\ \lambda_2 = \sqrt{(\mu_{2,0}(F) + \mu_{0,2}(F))/2 + \sqrt{4\mu_{1,1}(F)^2 + (\mu_{2,0}(F) - \mu_{0,2}(F))^2}}\end{gathered} \tag{28}$$

On the basis of the computed ellipse parameter, the candidate regions with the aspect ratio b_R/a_R in the interval of [1.0, 3.0] are retained as face regions.

3. Template Matching : The final stage of verification involves template matching where pixel intensity comparison between the pre-defined template face and the grayscale face candidate region is performed. The cross correlation is used as a similarity measure between the template and the face candidate. The template face is obtained by averaging frontal face image from CMU-PIE (Carnegie Mellon University Pose, Illumination, and Expression) dataset [56]. Initially, each remaining face candidate region is extracted from grayscale image using the binary mask from previous stage (Any holes within the binary mask are filled to retain eye and mouth features—Fig. 16c). The resultant image is transformed to grayscale for matching (Fig. 16d). Then grayscale template image T is aligned with considered candidate region (Fig. 16e) by: (i) resizing T according to the length of minor and major axis (a_R, b_R) estimated from previous stage, (ii) rotating by θ_R prior to place it on the centroid of the candidate region. Finally cross correlation is evaluated between aligned template and considered face candidate region. In our experiment, the cross correlation threshold is set to 0.6.

5.2 *Experimental Result*

We applied the scheme outlined in Sect. 5.1 to locate face region in images taken from Bao database. Figure 17 shows the procedure used to detect faces in a scene where

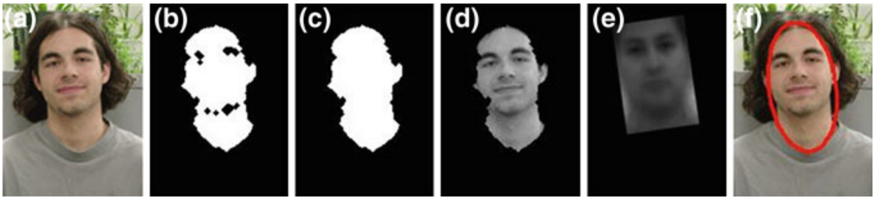

Fig. 16 Overview of template matching process

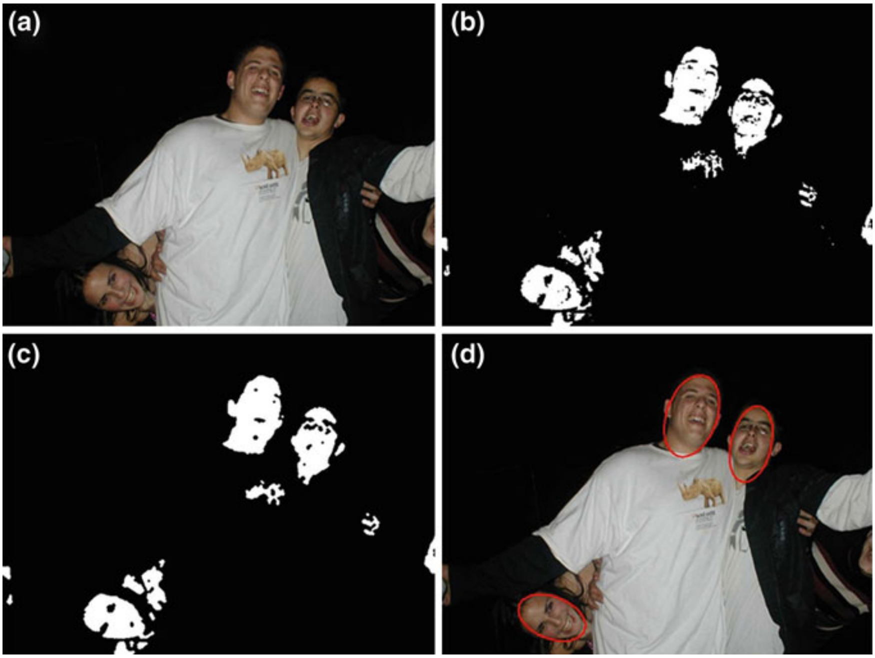

Fig. 17 Face detection sequence: **a** Original image, **b** Binary skin map obtained from skin color classification, **c** potential face candidates identified after morphological operations on binary skin map, **d** Final detection result after verification

three faces are successfully detected while other regions classified as skin are rejected from the verification stage. In particular, this example demonstrates the effectiveness of proposed feature-based detection solution in dealing with wider range of face poses than the detector outlined in Sect. 4, where the former successfully located face of almost 90° rotated.

Another strength of this feature-based detection system is its computational speed. Compared to the image-based solution outlined in Sect. 4 which requires exhaustive scanning of the image, the runtime of this solution is much faster since verification only takes place for identified face candidate regions. For example, the MATLAB implementation of the feature-based detection system took only 2146 seconds to process all 337 images from Bao dataset in the same system environment used in Sect. 4 (Table 5). It is worthwhile to point out that direct comparison of processing speed may not be meaningful as MATLAB implementations are not optimized. However, given the computational complexity of underlying workflow, it is reasonable to assume that feature-based face detection system is much faster even in real-world implementation. Figure 18 illustrates the robustness of the proposed scheme for faces of different ethnicity under various lighting conditions, including indoor and outdoor.

Although the proposed detector yields encouraging results, there are certain types of challenging condition that limit the performance of the system.

- The detector becomes unstable if other part of body (e.g. neck or hand) is placed in contact with a face (Fig. 19a), or if face overlaps with other skin-colored region (e.g. background objects or other faces) in the line of camera's sight (Fig. 19b). In this case, it's unable to separate each object into distinct clusters during face

Table 5 Performance comparison between two face detection systems outlined in Sects. 4 and 5 on 337 images of Bao dataset

Face detection method	No. FP	Detection rate (%)	Processing delay (s)
Hybrid method in Sect. 4	64	94.6	14286
Feature-based method in Sect. 5	318	78.3	2146

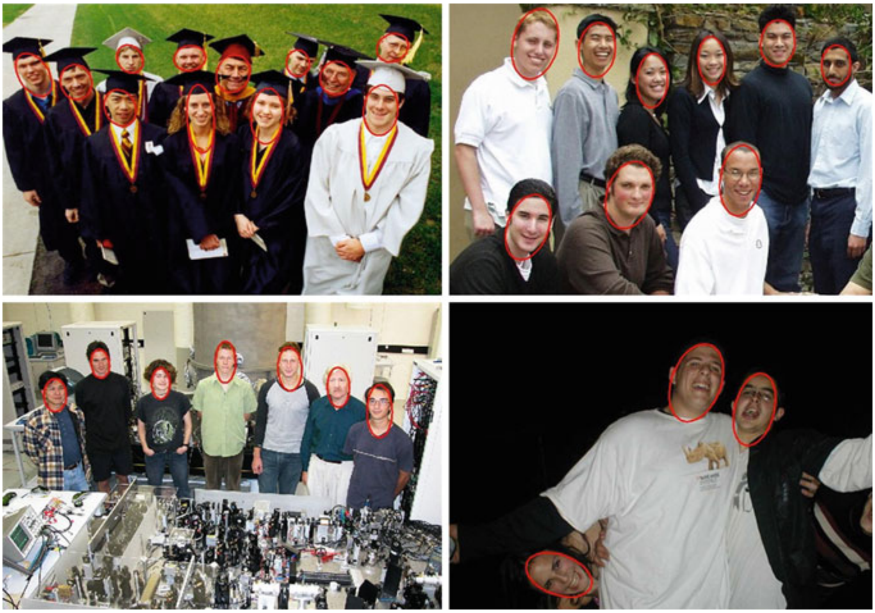

Fig. 18 Examples of the successful detection of faces with different skin colors and poses

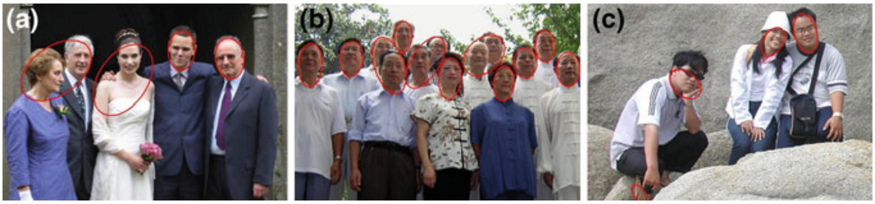

Fig. 19 Examples of the failed detection of faces

candidate identification and verify them individually. This particular scenario often results in generation of false positives or false negatives.

- Template matching tends to be highly tolerable to false positives (e.g. skin-colored background objects, and other part of skin-colored body) and thus, often fails to properly filters out other part of bodies with skin color. (Fig 19c)

Due to aforementioned limitations, face detection system introduced in this section yields suboptimal detection accuracy compared to image-based approach from the last section (See Table 5). Therefore, there exists a tradeoff between computational efficiency and detection accuracy. Following modification can be considered in order to enhance the detection performance of the feature-based detector:

- The true positive rate can be improved by properly segmenting face regions when face is presented in contact with other skin area, or presented in front of skin-colored background during face candidate identification stage. This requires use of other modalities, such as shape or texture, since color cue itself is not sufficient to distinguish between them.
- The number of false positives can be decreased by introducing more constraints into verification stage. For example, [30, 58] impose pose angle constraints that if the vertical orientation of face region is beyond commonly accepted interval, the corresponding region is rejected as non-face.

6 Conclusion

The face plays a crucial role in our human social interaction; face conveys people's identity and facial expression can be used as an important means of communication. Therefore, it is not surprising that human facial analysis has been considered as one of the most active areas of research in digital image processing and computer vision applications. While detection of faces is one of the basic tasks for human, it is not trivial task in computer vision, since faces have a high degree of variation in shape, color, and texture depending on imaging condition. In order to build a reliable and robust face detection system, several cues, such as motion, shape, color, and texture have been taken into account. Among available cues, color cue is advantageous due to its low computational complexity, high discriminative power, and robustness against geometrical transformation under stable illumination condition.

This chapter addressed several frequently encountered issues when using skin color as a feature for face detection, such as: (i) selection of the suitable color representation (i.e. colorspace) to perform color classification, (ii) selection of modeling scheme to represent the skin color distribution, (iii) dealing with its dependence to the illumination condition, (iv) how to apply skin color classification results in high-level analysis system.

Following conclusions can be drawn from this chapter:

1. Skin color analysis for face detection application involves a pixel-wise classification to discriminate skin and non-skin pixels. In order to select an optimal combination of color representation and skin color modeling scheme, the desired performance requirements (in terms of accuracy and processing speed), the available computational resources, as well as the amount of data available for training should be considered.
2. Color constancy algorithms can improve skin pixel classification performance by compensating the effect of scene illuminant in the recorded image and revealing the underlying color of object. In this chapter, we mainly reviewed low-complexity solutions, suitable to be deployed as a pre-processor for face detection framework. Experimentation performed on standard color face database demonstrates that face detection accuracy can be enhanced by applying them prior to skin color analysis.
3. Skin color cue can be utilized in face detection system in following two ways: (i) In image-based face detection approach, the faster and more accurate exhaustive face search can be achieved by using skin color with the purpose of guiding the search. Pixel-level classification of skin and non-skin is sufficient for this purpose, (ii) In feature-based face detection approach, color provides visual cue to focus attention in the scene by identifying a set of skin-colored regions that may contain face objects. Typically, local spatial context of skin pixel distribution is considered after skin color classification by exploiting morphological operations and connected component analysis.

References

1. Albiol A, Torres L, Delp E (2001) Optimum color spaces for skin detection. In: Proceedings of international conference on image processing, Thessaloniki, vol 1, pp 122–124
2. Barnard K, Cardei V, Funt B (2002) A comparison of computational color constancy algorithms. I: methodology and experiments with synthesized data. IEEE Trans Image Process 11(9):972–984
3. Bilal S, Akmeliawati R, Salami M, Shafie A (2012) Dynamic approach for real-time skin detection. J Real-Time Image Process pp 1–15
4. Brown D, Craw I, Lewthwaite J (2001) A SOM based approach to skin detection with application in real time systems. In: Proceedings of the British machine vision conference, University of Manchester, UK, pp 491-500

5. Buchsbaum G (1980) A spatial processor model for object colour perception. J Franklin Inst 310(1):1–26
6. Caetano T, Olabarriaga S, Barone D (2002) Performance evaluation of single and multiple-Gaussian models for skin color modeling. In: Proceedings XV Brazilian symposium on computer graphics and image processing, Brazil, 275–282
7. Chai D, Ngan KN (1998) Locating facial region of a head-and-shoulders color image. In: Proceedings of the international Conference on face and gesture recognition, pp 124–129
8. Chaves-Gonz+-lez JM, Vega-Rodr+-guez MA, G+-mez-Pulido JA, S+-nchez-P+-rez JM, (2010) Detecting skin in face recognition systems: a colour spaces study. Digital Sig Proc 20(3):806–823
9. Chen HY, Huang CL, Fu CM (2008) Hybrid-boost learning for multi-pose face detection and facial expression recognition. Pattern Recogn 41(3):1173–1185
10. Conci A, Nunes E, Pantrigo JJ, Sánchez A (2008) Comparing color and texture-based algorithms for human skin detection. In: Computer interaction 5:166–173
11. Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. J R Stat Soc, Series B 39(1):1–38
12. Erdem C, Ulukaya S, Karaali A, Erdem A (2011) Combining Haar feature and skin color based classifiers for face detection. In: IEEE international conference on acoustics, speech and signal processing, pp 1497–1500
13. Fasel I, Fortenberry B, Movellan J (2005) A generative framework for real time object detection and classification. Comput Vis Image Underst 98(1):182–210
14. Fawcett T (2006) An introduction to ROC analysis. Pattern Recogn Lett 27(8):861–874
15. Finlayson GD, Trezzi E (2004) Shades of Gray and Colour Constancy. In: Twelfth color imaging conference: color science and engineering systems, technologies, and applications, pp 37–41
16. Frischholz R (2008) Bao face database at the face detection homepage. http://www.facedetection.com. Accessed 07 Dec 2012
17. Fritsch J, Lang S, Kleinehagenbrock M, Fink G, Sagerer G (2002) Improving adaptive skin color segmentation by incorporating results from face detection. In: Proceedings of the IEEE international workshop on robot and human interactive communication, pp 337–343
18. Fu Z, Yang J, Hu W, Tan T (2004) Mixture clustering using multidimensional histograms for skin detection. In: Proceedings international conference on pattern recognition, vol 4. Brighton, 549–552
19. Gehler P, Rother C, Blake A, Minka T, Sharp T (2008) Bayesian color constancy revisited. http://www.kyb.tuebingen.mpg.de/bs/people/pgehler/colour/. In: IEEE conference on computer vision and pattern recognition, pp 1–8
20. Gevers T, Smeulders AW (1999) Color-based object recognition. Pattern Recognit 32(3):453–464
21. Gevers T, Gijsenij A, van de Weijer J, Geusebroek JM (2012) Pixel-based photometric invariance, Wiley Inc., pp 47–68
22. Gijsenij A, Gevers T, van de Weijer J (2011) Computational color constancy: survey and experiments. IEEE Trans Image Process 20(9):2475–2489
23. Gomez G, Morales EF (2002) Automatic feature construction and a simple rule induction algorithm for skin detection. In: Proceedings of the ICML workshop on machine learning in computer vision, pp 31–38
24. Greenspan H, Goldberger J, Eshet I (2001) Mixture model for face-color modeling and segmentation. Pattern Recogn Lett 22(14):1525–1536
25. Hadid A, Pietikäinen M (2006) A hybrid approach to face detection under unconstrained environments. In: International conference on pattern recognition, vol 1:227–230
26. Hadid A, Pietikäinen M, Ahonen T (2004) A discriminative feature space for detecting and recognizing faces. In: Proceedings IEEE Conference on computer vision and pattern recognition, vol 2, pp II-797–II-804
27. Hanbury A (2003) Circular statistics applied to colour images. In: Proceedings of the computer vision winter workshop, Valtice, pp 55–60

28. Haralick R, Shapiro L (1992) Computer and robot vision, addison-wesley longman publishing Co Inc, 1st edn. vol 1, Boston
29. Hassanpour R, Shahbahrami A, Wong S (2008) Adaptive Gaussian mixture model for skin color segmentation. Eng Technol 31(July):1–6
30. Herodotou N, Plataniotis KN, Venetsanopoulos AN (2000) Image Processing Techniques for Multimedia Processing. In: Guan L, Kung SY, Larsen J (eds) Multimedia image and video processing. CRC Press, chap 5:97–130
31. Hjelmås E, Low BK (2001) Face detection: a survey. Comput Vis Image Underst 83(3):236–274
32. Hossain MF, Shamsi M, Alsharif MR, Zoroofi RA, Yamashita K (2012) Automatic facial skin detection using gaussian mixture model under varying illumination. Int J Innov Comput I 8(2):1135–1144
33. Hsu RL, Abdel-Mottaleb M, Jain A (2002) Face detection in color images. IEEE Trans Pattern Anal Mach Intell 24(5):696–706
34. Jensen OH (2008) Implementing the viola-jones face detection algorithm. Technical university of Denmark, department of informatics and mathematical modeling, master's thesis, Denmark
35. Jin H, Liu Q, Lu H, Tong X (2004) Face detection using improved LBP under bayesian framework. In: Proceeding of the international conference on image and graphics, pp 306–309
36. Jones MJ, Rehg JM (2002) Statistical color models with application to skin detection. Int J Comput Vision 46(1):81–96
37. Jun B, Kim D (2012) Robust face detection using local gradient patterns and evidence accumulation. Pattern Recognit 45(9):3304–3316
38. Kakumanu P, Makrogiannis S, Bourbakis N (2007) A survey of skin-color modeling and detection methods. Pattern Recognit 40(3):1106–1122
39. Kawulok M (2008) Dynamic skin detection in color images for sign language recognition. In: Elmoataz A, Lezoray O, Nouboud F, Mammass D (eds) ICISP Lecture notes in computer science, vol 5099. Springer, France, 112–119
40. Khan R, Hanbury A, Sablatnig R, Stöttinger J, Khan F, Khan F (2012 a) Systematic skin segmentation: merging spatial and non-spatial data. Multimed tools appl pp 1–25
41. Khan R, Hanbury A, Stöttinger J, Bais A (2012 b) Color based skin classification. Pattern Recognit Lett 33(2):157–163
42. Kovac J, Peer P, Solina F (2003) Human skin color clustering for face detection. In: The IEEE region 8 EUROCON 2003 computer tool, vol 2:144–148
43. von Kries J (1970) Influence of adaptation on the effects produced by luminous stimuli. In: MacAdam D (ed) Sources of color science, MIT Press, pp 109–119
44. Land EH (1977) The retinex theory of color vision. Sci Am 237(6):108–128
45. Lee JY, Yoo SI (2002) An elliptical boundary model for skin color detection. In: Proceedings international conference on imaging science, systems, and technology. pp 81–96
46. Lienhart R, Maydt J (2002) An extended set of Haar-like features for rapid object detection. In: Proceedings international confrence on image processing, vol 1, pp 900–903
47. Louis W, Plataniotis KN (2011) Co-occurrence of local binary patterns features for frontal face detection in surveillance applications. EURASIP J Image Video Proces 2011
48. Moon H, Chellappa R, Rosenfeld A (2002) Optimal edge-based shape detection. IEEE Trans Image Process 11(11):1209–1227
49. Naji SA, Zainuddin R, Jalab HA (2012) Skin segmentation based on multi pixel color clustering models. Digital Sig Process 22(6):933–940
50. Ojala T, Pietik+inen M, Harwood D, (1996) A comparative study of texture measures with classification based on featured distributions. Pattern Recognit 29(1):51–59
51. Phung S, Bouzerdoum SA, Chai SD (2005) Skin segmentation using color pixel classification: analysis and comparison. IEEE Trans Pattern Anal Mach Intell 27(1):148–154
52. Plataniotis KN, Venetsanopoulos AN (2000) Color image processing and applications. Springer-Verlag, New York
53. Schmugge SJ, Jayaram S, Shin MC, Tsap LV (2007) Objective evaluation of approaches of skin detection using ROC analysis. Comput Vis Image Underst 108:41–51

54. Schwartz WR, Gopalan R, Chellappa R, Davis LS (2009) Robust human detection under occlusion by integrating face and person detectors. In: Proceedings international conference on advances in biometrics, pp 970–979
55. Sigal L, Sclaroff S, Athitsos V (2004) Skin color-based video segmentation under time-varying illumination. IEEE Trans Pattern Anal Mach Intell 26(7):862–877
56. Sim T, Baker S, Bsat M (2003) The CMU pose, illumination, and expression database. IEEE Trans Pattern Anal Mach Intell 25(12):1615–1618
57. Smith AR (1978) Color gamut transform pairs. SIGGRAPH Comput Graph 12(3):12–19
58. Sobottka K, Pitas I (1996) Extraction of facial regions and features using color and shape information. In: Proceedings international conference of pattern recognition, pp 421–425
59. Soriano M, Martinkauppi B, Huovinen S, Laaksonen M (2000) Skin detection in video under changing illumination conditions. In: Procedings international conference on pattern recognition, vol 1:839–842
60. Störring M (2004) Computer vision and human skin colour: a Ph.D. Computer vision and media technology laboratory. Dissertation, Aalborg University
61. Sun HM (2010) Skin detection for single images using dynamic skin color modeling. Pattern Recognit 43(4):1413–1420
62. Terrillon JC, David M, Akamatsu S (1998) Automatic detection of human faces in natural scene images by use of a skin color model and of invariant moments. In: Proceedings IEEE international conference on automatic face and gesture recognition, pp 112–117
63. Terrillon JC, Shirazi M, Fukamachi H, Akamatsu S (2000) Comparative performance of different skin chrominance models and chrominance spaces for the automatic detection of human faces in color images. In: Proceedings of the IEEE international conference on automatic face and gesture recognition, pp 54–61
64. Terrillon JC, Pilpre A, Niwa Y, Yamamoto K (2003) Analysis of a large set of color spaces for skin pixel detection in color images. In: Internationa Conference on quality control by artificial vision, vol 5132, pp 433–446
65. Vezhnevets V, Sazonov V, Andreeva A (2003) A survey on pixel-based skin color detection techniques. In: Proceedings of the GRAPHICON-2003, pp 85–92
66. Viola M, Jones MJ, Viola P (2003) Fast multi-view face detection. In: Proceedings of the computer vision and pattern recognition
67. Viola P, Jones MJ (2004) Robust real-time face detection. Int J Comput Vision 57(2):137–154
68. Wang X, Xu H, Wang H, Li H (2008) Robust real-time face detection with skin color detection and the modified census transform. In: International confrence on information and automation, pp 590–595
69. Wang X, Zhang X, Yao J (2011) Skin color detection under complex background. In: International conference on mechatronic science, electric engineering and computer (MEC), pp 1985–1988
70. Wei Z, Dong Y, Zhao F, Bai H (2012) Face detection based on multi-scale enhanced local texture feature sets. In: IEEE International conference on acoustics, speech and signal processing, pp 953–956
71. van de Weijer J, Gevers T, Gijsenij A (2007) Edge-based color constancy. IEEE Trans Image Process 16(9):2207–2214
72. Yang G, Huang TS (1994) Human face detection in a complex background. Pattern Recognit 27(1):53–63
73. Yang J, Lu W, Waibel A (1997) Skin-color modeling and adaptation. In: Proceedings of the Asian conference on computer vision-volume II, Springer-Verlag, pp 687–694
74. Yang MH, Ahuja N (1999) Gaussian mixture model for human skin color and its applications in image and video databases. In: Proceedings of the SPIE, pp 458–466
75. Yang MH, Kriegman DJ, Ahuja N (2002) Detecting faces in images: a survey. IEEE Trans Pattern Anal Mach Intell 24(1):34–58
76. Yendrikhovskij SN, Blommaert FJJ, de Ridder H (1999) Color reproduction and the naturalness constraint. Color Res Appl 24(1):52–67

77. Zarit BD, Super BJ, Quek FKH (1999) Comparison of five color models in skin pixel classification. In: Proceedings of the International workshop on recognition, analysis, and tracking of faces and gestures in real-time systems, pp 58–63
78. Zeng H, Luo R (2012) A new method for skin color enhancement. In: Proceedings of the SPIE, vol 8292. pp 82,920K.1–9
79. Zhang C, Zhang Z (2010) A survey of recent advances in face detection. Tech Rep MSR-TR-2010-66, Microsoft Research
80. Zhang L, Chu R, Xiang S, Liao S, Li S (2007) Face detection based on Multi-Block LBP Representation. In: Advances in biometrics, lecture notes in computer science, vol 4642. Springer-Verlag, pp 11–18
81. Zhao W, Chellappa R, Phillips PJ, Rosenfeld A (2003) Face recognition: a literature survey. ACM Comput Surv 35(4):399–458