Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

Real-Time Imaging 9 (2003) 277–287

www.elsevier.com/locate/rti

A novel method for detecting lips, eyes and faces in real time

Cheng-Chin Chiang*, Wen-Kai Tai, Mau-Tsuen Yang, Yi-Ting Huang, Chi-Jaung Huang

Department of Computer Science and Information Engineering, National Dong Hwa University, Shoufeng, Hualien 974, Taiwan, ROC

Abstract

This paper presents a real-time face detection algorithm for locating faces in images and videos. This algorithm finds not only the face regions, but also the precise locations of the facial components such as eyes and lips. The algorithm starts from the extraction of skin pixels based upon rules derived from a simple quadratic polynomial model. Interestingly, with a minor modification, this polynomial model is also applicable to the extraction of lips. The benefits of applying these two similar polynomial models are twofold. First, much computation time are saved. Second, both extraction processes can be performed simultaneously in one scan of the image or video frame. The eye components are then extracted after the extraction of skin pixels and lips. Afterwards, the algorithm removes the falsely extracted components by verifying with rules derived from the spatial and geometrical relationships of facial components. Finally, the precise face regions are determined accordingly. According to the experimental results, the proposed algorithm exhibits satisfactory performance in terms of both accuracy and speed for detecting faces with wide variations in size, scale, orientation, color, and expressions.

1. Introduction

In recent years, the fast advancement of the image processing techniques and the cost down of various image/video acquisition devices encouraged the development of many computer vision applications, such as vision-based surveillance, vision-based man–machine interfaces, vision-based biometrics, and so on. Among these many applications, face recognition is one of the central tasks that attract the attention of more and more researchers. A number of works in the literature had presented some face recognition applications in laboratorical and commercial scales [1–11]. One of the important tasks in designing a good face recognition system is the design of an efficient algorithm to locate faces in captured images or video. Actually, face detection is also a central task in some applications other than recognition systems. For example, in some video transmission applications, human faces are the only changing foreground objects in the video frames. Therefore, the repeated encoding, transmission and decoding of unchanged background parts are avoided to save the network bandwidth and computations. Hence, face detection plays a key role in segmenting the faces from the video background. As the face detection is always the first step in the processes of these recognition or transmission systems, its performance would put a strict limit on the achieved performance of the whole system. Ideally, a good face detector should accurately extract all faces in images regardless of their positions, scales, orientations, colors, shapes, poses, expressions and light conditions. However, for the current state of the art in image processing technologies, this goal is a big challenge. For this reason, many designed face detectors deal with only upright and frontal faces in well-constrained environments [1,12–16].

In addition to the accuracy, another important concern is the detecting speed. For instances, in many video phones and surveillance applications, the real-time speed is a critical requirement. This real-time speed requirement prohibited many algorithms that precisely extract faces at the cost of an extensive amount of computation time. Some high-speed computer CPUs may provide a good hardware solution to the speed requirement; however, the high costs of these powerful CPUs may also cut down the acceptability of these systems for common users.

In this paper, we propose a novel real-time face detection algorithm that can accurately locate both the face regions in images and the eyes and lips for each

*Corresponding author. Fax: +886-38631040.
E-mail address: ccchiang@mail.ndhu.edu.tw (C.-C. Chiang).

1077-2014/$ - see front matter © 2003 Elsevier Ltd. All rights reserved.
doi:10.1016/j.rti.2003.08.003

located face. The detailed capability specifications of the proposed algorithm are described as follows:

1. Users can tilt their faces left or right for about 45°.
2. Users can raise, lower, or rotate their heads as long as neither lips nor eyes are occluded.
3. The sizes of faces are limited to the size between 1600 ($= 40 \times 40$) pixels and 9216 ($= 96 \times 96$) pixels. This limitation is set to fit the resolution requirements for general face recognition engines. The values can be easily adjusted if different resolutions are demanded.

We assume that the environment light uniformly illuminates on the faces. That is, we exclude the cases that light is focused on partial areas of face images. The basic concept of the proposed algorithm is to extract and then verify the desired components, including skins, lips, eyes and faces with several simple rules. It is found that the defined rules can handle a large degree of variations in faces. Due to the simplicity and effectiveness of these rules, the proposed algorithm can accurately detect faces with wide variations at a real-time speed.

The rest of the paper is organized as follows. Section 2 makes a brief survey on some related work. The details of the proposed algorithm are presented in Section 3. In order to show the effectiveness of the proposed algorithm, some experimental results are provided in Section 4. The performance evaluation and comparisons in terms of the accuracy and speed is also given there. Finally, we conclude this paper in Section 5.

2. Related works

A straightforward approach for detecting faces in images is through template correlation matching [1–4]. The template can be designed or learned through the collection of a set of face patterns. During the matching process, a template is convolved with the subimages everywhere within the input image to find the possible candidates based on a predefined similarity or distance. To handle the possible variations in size, orientation, and shape, etc., two methods are usually adopted. The first is to resize each input image to different dimensions before matching. Then, the template matching is repeatedly performed on every resized input image. The alternative is to use multiple face templates with different variations in size, expression, orientation or lighting for matching with the input image. Obviously, no matter what way is adopted, the matching time would increase drastically with the numbers and the dimensions of the used templates and input images.

To reduce the searching time of the possible face candidates over the input images, a process of skin pixel extraction is commonly used [17–33]. The basic idea of the skin pixel extraction is to build a statistic model for colors of skin-like pixels. For example, the mixture of Gaussian distributions is a useful probabilistic model for this purpose [17,20,26,29]. The probabilistic model is used to calculate the probability of being a skin pixel for each pixel. A threshold is then set on this probability to remove those pixels that are very unlikely to be skin pixels. Some other researchers proposed the neural network approach to approximate more complex models for skin pixels [16,30]. However, the increased complexities of these neural models increase also the computation costs.

With the identified skin pixels within images, the connected skin pixels form several candidate regions for faces. Accordingly, the prestored face templates are used to match the images or subimages within these candidate regions [34,35]. Afterwards, those redundant skin regions that do not contain face-like patterns are discarded. In addition to the template matching, the neural network verifier is an alternative design for face pattern verifications [12]. According to the experimental results, the neural network verifier usually exhibits higher variation tolerance than the template matching. However, the long period of training time and the requirement of large training sets are the major drawbacks of this kind of verifiers.

The above face detection method can be referred as a top-down (or image-based) detection. The term "top-down" means that the face regions are determined without resorting to the identification of individual facial components such as eyes, noses and lips, etc. In this paper, we present an alternative approach, which we called the bottom-up (or feature-based) detection, to locate face regions. The term "bottom-up" means that the precise face regions are constructed from the identified facial components. Namely, the facial components must be extracted prior to the determination of the precise locations of face regions. The design philosophy of this approach is that some individual facial components like eyes and lips usually exhibit visual properties that are less sensitive to the face variations mentioned previously. For example, the eyes and lips are usually the darker components and the dark-skin components on normal faces, respectively. These properties enable human eyes to recognize the facial components very easily. Thus, the extraction of these facial components is generally more stable. In addition, since the individual components are much smaller than the whole face in size, the processing time required during extraction is thus also much shorter.

3. Rule-based face detection algorithm

According to the framework of the bottom-up detection approach, the proposed algorithm is designed to extract the facial components including lips and eyes.

In order to reducing the searching areas in the input images, the proposed algorithm also performs the extraction of skin pixels. However, instead of using probabilistic models, we use a quadratic polynomial model for the color model of skin pixels to reduce the computation time. Moreover, we also extend this polynomial model to the extraction of lip components with a minor modification. Finally, the falsely extracted eye and lip components are removed based on a set of rules induced from the common spatial and geometrical relationships among normal facial components. The final precise face regions are then determined accordingly.

3.1. Rules for skin-color region extraction

The purpose of extracting skin-color regions is to reduce the searching time for possible face regions on the input image. In order to alleviate the influence of environment light brightness on extracting skin pixels, the proposed algorithm adopts the chromatic color coordinate for color representation. For chromatic color space, each pixel is represented by two values, denoted by (r, g). The conversion from conventional RGB color space to chromatic color space is defined as follows:

$$r = \frac{R}{R+G+B},$$
$$g = \frac{G}{R+G+B},$$

where R, G, B denotes the intensities of pixels in red, green and blue channels, respectively. According to the above transformation, it is easy to see that the brightness variation in images can be normalized in chromatic color space. Moreover, the color representation in this two-dimensional chromatic color space enables us to visualize and analyze very easily when building the color model for skin pixels. To build a color model for skin pixels, the distribution of skin pixels over the $r-g$ plane is plotted in Fig. 1. Apparently, the skin pixels form a compact region over the $r-g$ plane. In many existing methods, the distribution is commonly approximated with some probabilistic models. For example, Gaussian mixture [16] model is a very popular one. The major drawback of probabilistic modeling is the high computation costs in calculating the probabilities. The probability calculation must be done for each pixel in the image. The calculation time would increase as the image size increases. As the objective of our algorithm is to detect faces in real-time, these computation costs are more or less obstructive to our goal. Hence, instead of trying to build a probabilistic model, we adopt a much simpler method that requires only very low computation costs. We modify the approach of Soriano and Martinkauppi [33], which uses two quadratic polynomials to approximate the upper and lower boundaries of the compact region, which are referred as *skin locus*, formed by the skin pixel distribution on the $r-g$ plane. These two polynomials are:

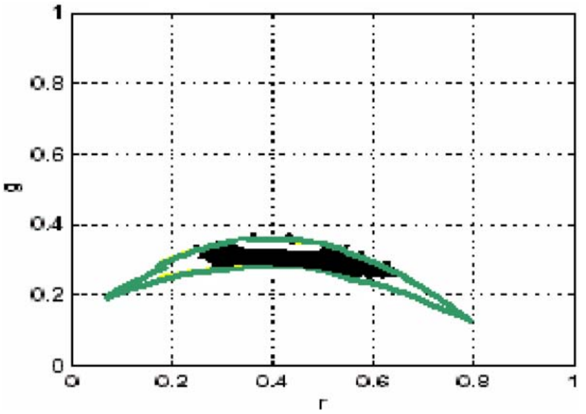


Fig. 1. Distribution of skin-color pixels on $r-g$ plane of chromatic color space and the upper and lower parabolic boundaries defined by the two quadratic polynomials [33].

$$f_{upper}(r) = -1.3767r^2 + 1.0743r + 0.1452, \tag{1}$$

$$f_{lower}(r) = -0.776r^2 + 0.5601r + 0.1766. \tag{2}$$

In fact, the coefficients of these two polynomials can be easily estimated by least mean square (LMS) error minimization. After drawing the boundaries for the compact region, we assume $(r_1, g_1), (r_2, g_2), \ldots, (r_n, g_n)$ to be n sampled points on the upper boundary. Then, we can get the following set of linear equations:

$$\begin{aligned} a_u r_1^2 + b_u r_1 + c_u &= g_1, \\ a_u r_2^2 + b_u r_2 + c_u &= g_2, \\ &\vdots \\ a_u r_n^2 + b_u r_n + c_u &= g_n, \end{aligned}$$

where a_u, b_u and c_u are the three coefficients of the quadratic polynomial for the upper boundary. Rewriting these equations as a matrix form, we get

$$\mathbf{Ru} = \mathbf{G}, \tag{3}$$

where

$$\mathbf{R} = \begin{bmatrix} r_1^2 & r_1 & 1 \\ r_2^2 & r_2 & 1 \\ \vdots & \vdots & \vdots \\ r_n^2 & r_n & 1 \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} a_u \\ b_u \\ c_u \end{bmatrix} \quad \text{and} \quad \mathbf{G} = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{bmatrix}.$$

Therefore, the coefficient vector $\mathbf{u}$ is computed according to

$$\mathbf{u} = \mathbf{R}^{\dagger}\mathbf{G},$$

where $\mathbf{R}^{\dagger}$ is the pseudoinverse of $\mathbf{R}$, which is equal to $(\mathbf{R}^t\mathbf{R})^{-1}\mathbf{R}^t$. Similarly, the coefficients of the quadratic polynomial for the lower boundary can be derived in this way. With the determined coefficients for the two polynomials, the skin-color pixels, which fall within the area between these two polynomials, can be extracted by the following rule checking on their r, g values:

$R1$: $g > f_{lower}(r)$ and $g < f_{upper}(r)$.

In Fig. 1, we see the above two inequalities define a crescent-like area on the $r - g$ plane. Besides these two polynomials, Soriano and Martinkauppi also defined a circle on the $r - g$ plane to exclude the bright white pixels, which falling around the point $(r, g) = (0.33, 0.33)$, from the region enclosed by the two polynomials. Therefore, the exclusive circle is defined by

$R2$: $W = (r - 0.33)^2 + (g - 0.33)^2 \leqslant 0.0004$.

Fig. 2 illustrates the result of applying rules $R1$ and $R2$ simultaneously for skin pixel extraction. According to our experiments, the above two rules are still not accurate enough to filter out the non-skin pixels, particularly for yellow-green, blue and orange pixels that fall around the top, left end and right end of the crescent area respectively.

To improve the extraction results, the proposed algorithm introduces two extra simple rules. These two rules are:

$R3$: $R > G > B$,

$R4$: $R - G \geqslant 45$.

Rule $R3$ is derived from the observation that the skin pixels tend to be red and yellow. This observation implies that the blue intensity is always the smallest one among the three channels. With Rule $R3$, the pixels tending to be blue can be effectively removed. Rule $R4$ is defined in order to remove the yellow-green pixels. Fig. 3 illustrates the improved result after introducing Rules $R3$ and $R4$. It is clear to see that Rules $R1$ through $R4$ involve only very simple computations. The extraction process is much more efficient than those probabilistic approaches. In summary, we summarize the final rule for extracting skin-color pixels as follows:

$$S = \begin{cases} 1 & \text{if all } R1, R2, R3 \text{ and } R4 \text{ are true,} \\ 0 & \text{otherwise.} \end{cases}$$

where $S = 1$ means the examined pixel is a skin pixel. After the skin pixel extraction, we have to group the connected skin pixels into compact skin-color regions. To improve the compactness of the formed skin-color regions, the morphological dilation and erosion operations are performed over the extracted skin pixels. Then, the algorithm finds the bounding boxes for the connected components of skin pixels. These bounding boxes become the candidate skin-color regions for further identifications of individual facial components.

3.2. Rules for lips and eyes detection

Observing the pixels of lips, we find that the colors of lips range from dark red to purple under normal light condition. From the perspective of human visual perception, the lips are very easy to be differentiated from the face skins for any races of people because of their different contrasts in color. Based on this observation, the color distribution of the lips and normal skins

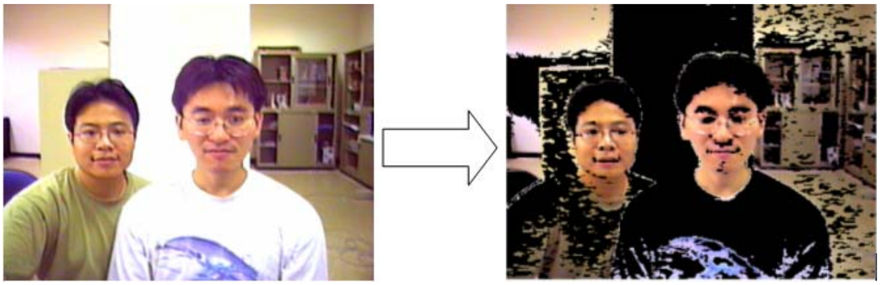


Fig. 2. Skin pixels filtering using the method of Soriano and Martinkauppi [33]. The left image is the input image and the right image is the resultant image.

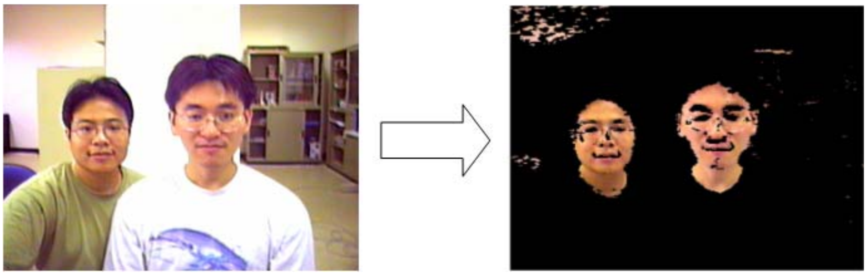


Fig. 3. Improved result of skin pixel extraction after using extra rules $R3$ and $R4$.

should be distinguishable. Actually, the experimental results show that these lip colors distribute at the lower areas of the crescent area defined by the skin colors on the $r-g$ plane. Similar to the extraction of skin pixels, we try to define another quadratic polynomial discriminant function for lip pixels. When defining the polynomial function, we have the following two design goals for achieving higher extraction speed:

1. The discriminant function should be computationally efficient, and
2. The detection of lip pixels can be done in parallel with the detection of skin pixels in one scan of the image or video frame.

Based on the above two guidelines and our observations about the distributions of lip colors, we find that the quadratic polynomial of the lower boundary, i.e. the $f_{lower}(r)$ defined in Eq. (2), can be reused. As the lip color distribute around the lower area of the crescent region of skin colors' distributions, we just need to slightly increase the value of the constant term in $f_{lower}(r)$ to get the upper boundary. Therefore, after the statistics on lip pixels, we define the two discriminant functions for lip pixels as

$$l(r) = -0.776r^2 + 0.5601r + 0.2123,$$
$$f_{lower}(r) = -0.776r^2 + 0.5601r + 0.1766. \tag{4}$$

During the discrimination using $l(r)$, the darker pixels on faces, which are usually the eyes, eyebrows or nose holes, also need to be excluded from the discrimination. We thus define the following rule for detecting lip pixels:

$$L = \begin{cases} 1 & \text{if } f_{lower}(r) \leqslant g \leqslant l(r) \text{ and } R \geqslant 20 \text{ and} \\ & \quad G \geqslant 20 \text{ and } B \geqslant 20, \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

where R, G, B are the intensity values in red, green and blue channels, respectively. $L=1$ indicates the examined pixel is a lip pixel. The reason of using RGB color space, instead of chromatic color space, to exclude dark pixels is that the chromatic color space is not suitable to distinguish between bright pixels and dark pixels. Both dark pixels and bright pixels might have the same converted $r-g$ values due to the normalization effect in brightness in chromatic color space.

The benefit of using the similar mathematical models in both skin detection and lips detection is that much computation time for the discriminant function can be saved. In addition, the detections of both skin pixels and lip pixels can be done in parallel in one scan of the image or video frame. Thus, this design completely follows the two design goals listed above. After detecting the lip pixels on the images, the proposed algorithm also perform the connected component labeling to group the connected lip pixels into candidate lip components.

The proposed algorithm does not deal with the faces that tilt left or right for more than 45°. Therefore, the lips are assumed to be always below the positions of the two eyes in a face. With this assumption, many impossible combinations for lips and eyes will be removed in later geometrical and spatial relationship verifications of facial components. For those other falsely extracted components, the algorithm will also remove them in the same component-based verifications process.

Considering the extraction of eye pixels, we observed that the eye components tend to be the darkest part on the faces under normal light condition, i.e. the light uniformly illuminates on faces. However, this claim does not imply that pixels of eyes are always black. Suppose that a dark eye pixel has RGB values of $(1,1,1)$. Then we get $r=g=0.333$ in chromatic color space. However, for any brighter white pixel on the images whose RGB values might be (v,v,v) for a larger value of v, the converted $r-g$ vector is $r=g=0.333$, too. Therefore, the polynomial discriminant function in chromatic color space is not suitable for extracting eye pixels. For this reason, the algorithm uses another simple way to extract the dark components on the faces. The eye components are extracted through a threshold operation (e.g. threshold $=20$) on the histogram-equalized grayscale image converted from the original color image. The bounding boxes of skin-color regions are used to remove the falsely extracted dark components. The method is found to be very efficient and effective [35]. Even though some dark components are wrongly detected, the component-based verification process coming next will remove them.

3.3. Rules for component verifications and face region determination

Among the extracted possible lip and eye components, there might be some false candidates. In order to remove these falsely extracted components, the proposed algorithm performs component verifications based on a set of rules derived from the common geometrical and spatial relationships among facial components. The rules used by the proposed algorithm are described in the following.

1. Let $\mathbf{p}_{eL}(x_{eL}, y_{eL})$ and $\mathbf{p}_{eR}(x_{eR}, y_{eR})$ be the two centers of the left eye component and right eye component, respectively. Then the angle between the line $\overline{\mathbf{p}_{eL}\mathbf{p}_{eR}}$ and the horizontal line must be in the range $[-45°, 45°]$. This rule is used to exclude those faces tilted left or right for more than 45°.
2. Without loss of generality, we assume $x_{eL} < x_{eR}$. Let $\mathbf{p}_m(x_m, y_m)$ be the center of the lip component and $\mathbf{p}_{mp}(x_{mp}, y_{mp})$ the point of projecting $\mathbf{p}_m(x_m, y_m)$ onto the line $\overline{\mathbf{p}_{eL}\mathbf{p}_{eR}}$. Given that the slope of $\overline{\mathbf{p}_{eL}\mathbf{p}_{eR}}$ is m_e, then the following spatial rules among lips and eyes

must be satisfied:

$$\begin{cases} x_{eL} \leqslant x_{mp} \leqslant x_{eR} & \text{if } |m_e| < 0.2, \\ x_{eL} \leqslant x_{mp} \leqslant x_{eR} + 0.15 * |x_{eR} - x_{eL}| & \text{if } 0.2 \leqslant m_e \leqslant 1, \\ x_{eL} - 0.15 * |x_{eR} - x_{eL}| \leqslant x_{mp} \leqslant x_{eR} & \text{if } -1 \leqslant m_e \leqslant -0.2. \end{cases}$$

The first rule implies that the projected position of the lip center must falls between the two eyes provided that the tilt angle of face is not large ($|m_e| < 0.2$). The second and the third rules exclude the cases that the lips position is not consistent with the tilt direction of heads. For example, the projected point of the lips cannot be at the left side of left eye, i.e., $x_{mp} \geqslant x_{eL}$, provided that the head tilts left ($0.2 \leqslant m_e \leqslant 1$), and vice versa.

3. Let $L(\overline{\mathbf{pq}})$ denote the length of the line segment $\overline{\mathbf{pq}}$ and the middle point of the two eyes is $\mathbf{p}_{ec}(x_{ec}, y_{ec})$. Then the following rule about the aspect ratio of face geometry must be satisfied:

$$0.8 \leqslant \frac{L(\overline{\mathbf{p}_m \mathbf{p}_{ec}})}{L(\overline{\mathbf{p}_{eL} \mathbf{p}_{eR}})} \leqslant 1.5.$$

4. Let $W(C_i)$ denote the width of the bounding box of component i. Then the following geometries rules about the facial components must be satisfied:

$$\frac{1}{1.3} \leqslant \frac{W(C_{Left\text{-}eye})}{W(C_{Right\text{-}eye})} \leqslant 1.3,$$

$$0.6 \leqslant \frac{W(C_{Left\text{-}eye})}{W(C_{Lip})} \leqslant 1.1 \quad \text{and} 0.6 \leqslant \frac{W(C_{Right\text{-}eye})}{W(C_{Lip})} \leqslant 1.1,$$

$$0.25 \leqslant \frac{W(C_{Left\text{-}eye})}{L(\overline{\mathbf{p}_{eL} \mathbf{p}_{eR}})} \leqslant 0.4 \quad \text{and} 0.25 \leqslant \frac{W(C_{Right\text{-}eye})}{L(\overline{\mathbf{p}_{eL} \mathbf{p}_{eR}})} \leqslant 0.4.$$

These rules are useful in filtering out the noise components. Note that the eyebrows sometimes connect to the eyes in the extracted components. For such cases, the algorithm regards the connected eye and eyebrow as a component. This action usually affects only a little precision for the located positions for eyes, but not the accuracy of the extracted eye components. The detection accuracy of faces is thus not significantly affected.

According to the verification on the above four geometrical and spatial relationship rules for eyes and lips, the candidate facial triangles on the skin-color regions can be extracted for further processing. In the proposed algorithm, the clear differentiation of lip components and eye components through Eq. (5) can greatly reduce the number of extracted triangles. In addition, the assumption that the lips are always below the eyes in one face is also very helpful in removing the impossible combinations, too.

3.4. The arbitration of confusing eye–lip triangles

Most noise components can be removed after the component verification process. However, it might be still possible that some confusing combinations need to be dealt with further. The confusing combinations result from those triangles that contain the lips from one face and the eyes from another. As shown in Fig. 4, we see three possible triangles T_1, T_2 and T_3 for facial components, where T_2 is an incorrect one. To solve the ambiguity, we propose a method to remove the confusing triangles. Firstly, a criterion, named skin color ratio (*SCR*), is defined for the arbitration of the confusing triangles. The *SCR* is defined as follows:

$$SCR(\Delta \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3) = \frac{N_{skin|\Delta \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3}}{A(\Delta \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3)}, \tag{6}$$

where $A(\Delta \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3)$ denotes the area of the triangle $\Delta \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3$ and $N_{skin|\Delta \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3}$ is the number of skin-color pixels within the triangle $\Delta \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3$. If *SCR* is very low, then $\Delta \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3$ is very likely to be a triangle that contains three facial components from distinct or scattered skin-color regions. For this kind of triangles, they are thus very possible to be unwanted because the higher ratio of non-skin pixels usually comes from the non-skin environment background that separates the contained multiple faces (refer to triangle $T2$ in Fig. 4). Hence, *SCR* is a good indication for the arbitration of confusing triangles.

In order to calculate $N_{skin|\Delta \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3}$, it might need a few computations. Fortunately, the introduced cost is not very high because the number of confusing triangles is usually very small after the component-based verifications. To save the computation time, we use a fast method to calculate $N_{skin|\Delta \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3}$. This method is to examine each detected skin pixel on the image to see if it is in the interior of the triangle $\Delta \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3$. If it is, then $N_{skin|\Delta \mathbf{p}_1 \mathbf{p}_2 \mathbf{p}_3}$ is increased by one. In this way, we need not to use time-consuming algorithms for tracing the

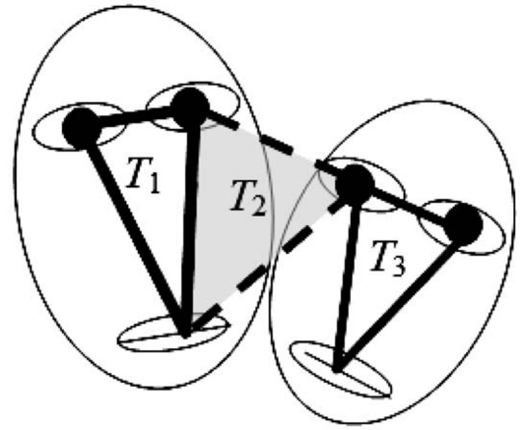


Fig. 4. Case of confusing triangles in facial component verifications. $T2$ is a bad confusing triangle.

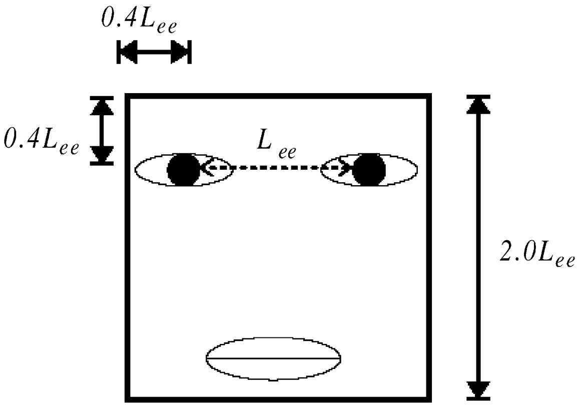


Fig. 5. Determination of a face region.

interior pixels of each confusing triangle. We just need to perform simple triangle interior checks over each confusing triangle for each skin pixel. According to the simple geometry mathematics, given the three vertices $\mathbf{p}_1(x_1, y_1)$, $\mathbf{p}_2(x_2, y_2)$, and $\mathbf{p}_3(x_3, y_3)$ of a triangle, the rule for the triangle interior check of any point $\mathbf{p}(x, y)$ is performed as follows:

$$C_1\colon\ x(y_2 - y_1) + y(x_1 - x_2) + (x_2y_1 - x_1y_2) > 0,$$
$$C_2\colon\ x(y_3 - y_2) + y(x_2 - x_3) + (x_3y_2 - x_2y_3) > 0,$$
$$C_3\colon\ x(y_1 - y_3) + y(x_3 - x_1) + (x_1y_3 - x_3y_1) > 0,$$

If(all C_1, C_2 and C_3 aretrue) or (all C_1, C_2 and C_3 are false) then $\mathbf{p}(x, y)$ is inside the triangle $\Delta\mathbf{p}_1\mathbf{p}_2\mathbf{p}_3$.

After extracting the triangles of facial components, the final face regions are simply determined according to the triangles. Fig. 5 shows the dimension of the bounding box for extracted face region. In Fig. 5, L_{ee} is the distance between the two eyes. L_{me} is the distance between the center point of the line segment connecting the two eyes and the center point of the extracted lips component. Note that the bounding box can be rotated according to tilted angle of the line connecting the two eyes.

4. Performance evaluation

For the performance evaluation, we have implemented the proposed algorithm on a PC with a Pentium III 800 CPU and 128M RAM. The implemented system has two modes of operations. The first is on-line mode that is designed to detect faces in video frames captured from a PC camera in real time. The other mode is off-line mode that is designed to detect faces in still images. To evaluate the accuracy and the speed of the proposed algorithm, we have prepared a test set that contains 1000 images. Among these 1000 images, 815 images with the dimension of 320 × 240 pixels are acquired by our ORITE VQ-681 USB PC camera while the other 185 images are collected from WWW. For the 815 images acquired by ourselves, they contain only Chinese people. In order to evaluate the robustness of the skin pixel extraction in our method, we collect the face images for many different races in the 185 images from WWW. Both of these two kinds of test images contain single- and multiple-face images and also enclose some variations in size, orientation, tilting and expression.

Table 1 lists the detection accuracy of the proposed algorithm. The average detection accuracy is 94.3% for our 815 images and 89.1% for the 185 images from WWW. This accuracy is comparable to most existing methods which detect only faces, but no lips and eyes. On evaluating the detection speed, we measure the number of faces detected per second. We use this as our speed measurement because our detection time taken to process each image frame depends on the number of faces in this image frame. Hence, the measurement of counting the number of image frames processed per second is not precise enough. According to our statistics on the 815 test images, our method achieves about 9 faces per second for image frames of dimension 320 × 240 pixels. Namely, for single-face images, we achieve a detection speed of 9 frames per second. In Table 2, we show the average percentage of the time taken by each process in the proposed algorithm. Our detection speed is roughly 10 times faster than Rowley–Baluja–Kanade detector [12]. In [36], Yang and Waibel presented a skin-color-based real-time face tracker with a detection speed of 10–30 frames per second. There detection speed is dependent on the distances between camera and users. Although their speed seems faster than ours, their algorithm does not locate the precise facial components of faces. That is, only the skin-color face regions are identified in their method. According to

Table 1
Detection accuracy of the proposed algorithm

	815 images captured by us	186 images from WWW
Total number of faces	1930	396
Number of correct detections	1820	353
Number of false detections	65	25
Detection accuracy	**94.3%**	**89.1%**

Table 2
Time percentage for the processes in the proposed algorithm

Process	**Average time percentage**
Skip/Lip pixel extraction	18%
Eye extraction	13%
Facial component verification	69%

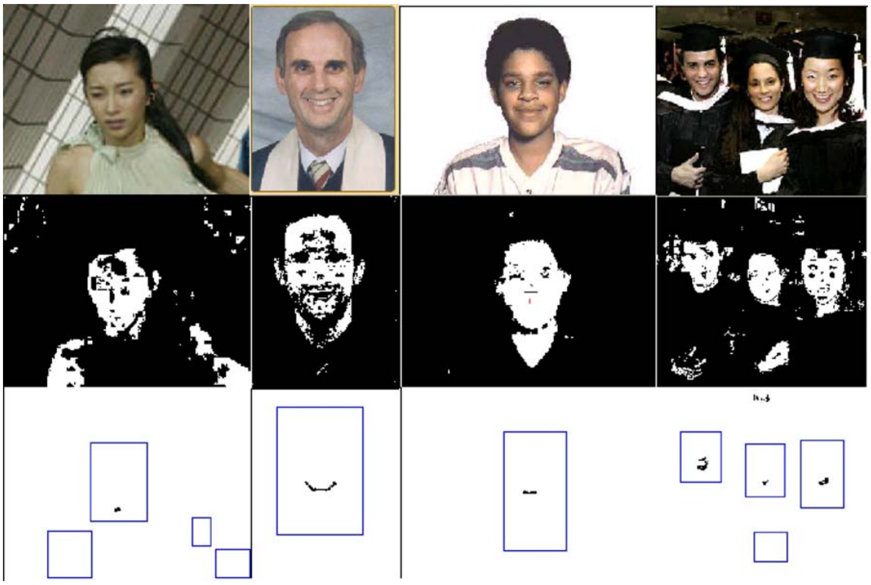


Fig. 6. Results of extracted skin-color pixels and lip pixels: the middle row is the extracted skin-color pixels. The bottom row is the extracted lip pixels. The rectangles in the bottom row indicate the bounding box for candidate face regions defined by skin-color pixels.

Tables 1 and 2, the proposed algorithm indeed achieves real-time detection of faces with a satisfactory accuracy. In Fig. 6, some snapshots of skin-color region detection are shown in the middle row. In these tested examples, we can see that the cases of faces in different races. It verifies the effectiveness and robustness of the proposed polynomial model for skin color. In Fig. 6, the results of detecting lip pixels are illustrated. The results show that the detections are very accurate and robust, too. In our experiments, the number of correct detection of lips is 2183 for the total of 2326 faces after performing the intersection of the extracted skin-color regions and the candidates for lip components. The number of false detection of lips is about 112 for the 1000 test images. The high accuracy of lips detection is very helpful to the component-based verification process coming next. It provides a strong support to the performance of the component verification process in terms of accuracy and speed. In Fig. 7, the extracted facial triangles composed of eyes and lips as well as the determined facial regions are shown for several tested images. These images contain cases of single/multiple persons, multiple orientations, body interference, face tilting, and different facial expressions. In Fig. 8, several failure cases are shown. According to our analysis, the major three reasons that causes the failure cases in the proposed method are

- *Occlusion of facial components*: The geometrical and spatial relationships among facial components are the basis of our verification process. If any eye component or lip component is missing due to body occlusion, head rotation, wearing, etc., then the proposed algorithm usually fails in the verification process.
- *Bad illuminating condition*: We assume that the light is in normal condition, i.e. the light is not color-biased and it uniformly illuminates on faces. If the assumptions are violated, then it causes the failure in the extraction of skin-color pixels and lip pixels. Particularly, the incorrect extraction of lip pixels always causes the wrong detection results.
- *Poor image quality*: Some of the images collected from WWW have poor quality because of the compression. Some of the facial features become distorted due to the compression, too.

To compare the performance of our detector with other existing method, we perform an experiment to test the same image set with other detector. To avoid the deviations caused by different implementation, we do not implement the detectors of other researchers by ourselves. Instead, we try to access the detectors publicly available on the WWW. Fortunately, we found the face detector developed by Henry Schneiderman and Takeo Kanade [37], who are with the Vision and Autonomous Systems Center (VSAC) of CMU, on the web page http://vasc.ri.cmu.edu/cgi-bin/demos/findface.cgi. This web page provides users the function to submit test images remotely. This detector is designed based on the image-based (or view-based) approach. That is, it finds only the face regions without identifying the precise facial components. This detector detects faces without using the color information. Therefore, the input images are in gray-level format. To test this detector, we submitted the 185 images collected from WWW to it. According to our statistics on the testing results, we list the performance of this detector and our detector in

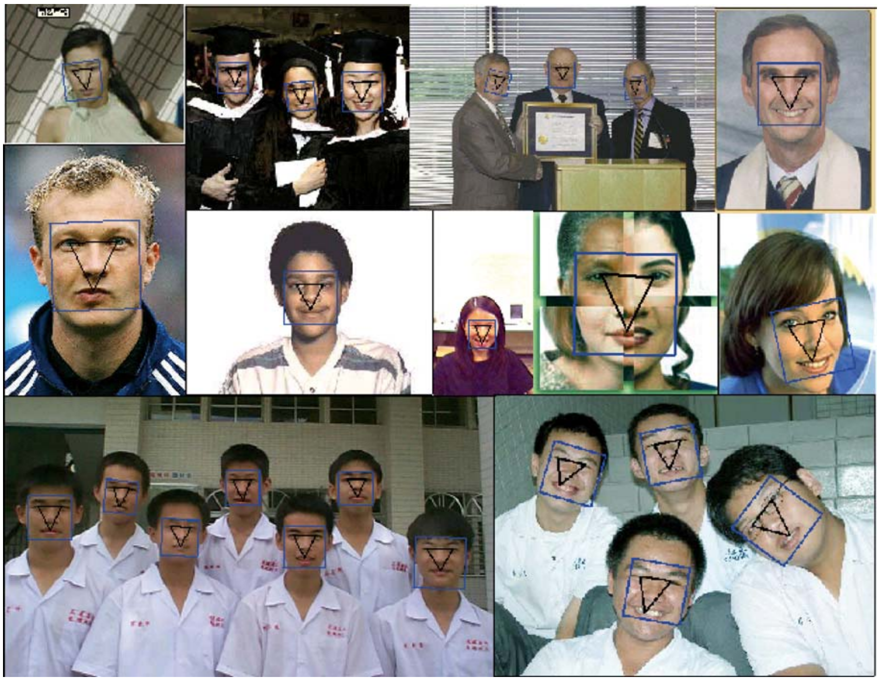
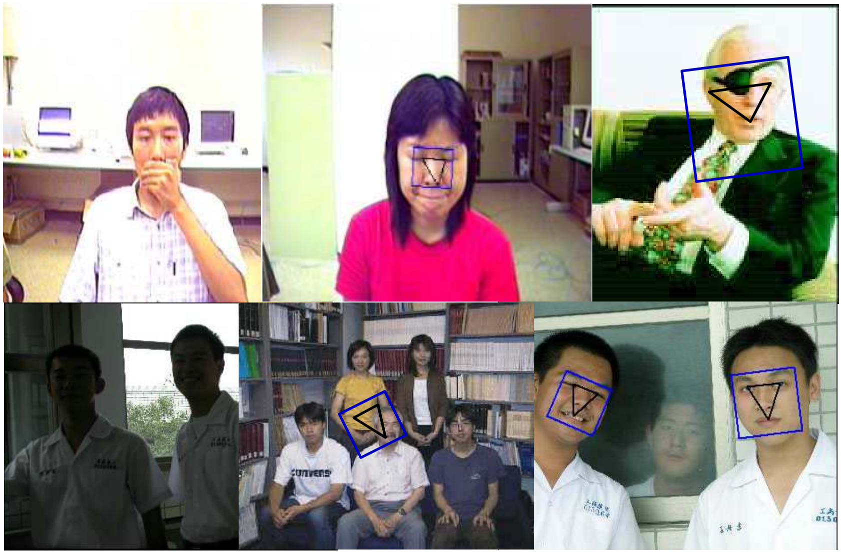


Fig. 7. Results of facial triangle detection and facial region determination.



a	b	c
d	e	f

Fig. 8. Failure cases for the proposed method: (a)–(c) failure caused by occlusion; (d) failure caused by bad illumination; (e) failure caused by poor image quality; (f) failure caused by bad color contrast.

Table 3 for comparison. Observing the detection results for CMU face detector, we find that this detector is more robust to the variations in illumination because it does not employ the color information in images. However, the major weakness of this detector is that it can not detect the tilted faces very well. In Fig. 9, we show some compared detection results for the two tested detectors.

In order to further test the robustness of the proposed algorithm, we also conduct the experiments of detecting faces wearing sunglasses. Fig. 10 demonstrates the results. The results show that the faces still can be correctly detected as long as the frame of the sunglasses is not very dark and thick. If the frame is too dark and thick, as shown in Fig. 8(c), the face is usually

Table 3
Performance comparison for the CMU detector developed by Henry Schneiderman and Takeo Kanadeand and our detector

Total number of faces = **396**	Number of correctly detected faces	Number of falsely detected faces	**Detection accuracy** (%)
CMU face detector	350	24	**88.4**
Our face detector	353	25	**89.1**

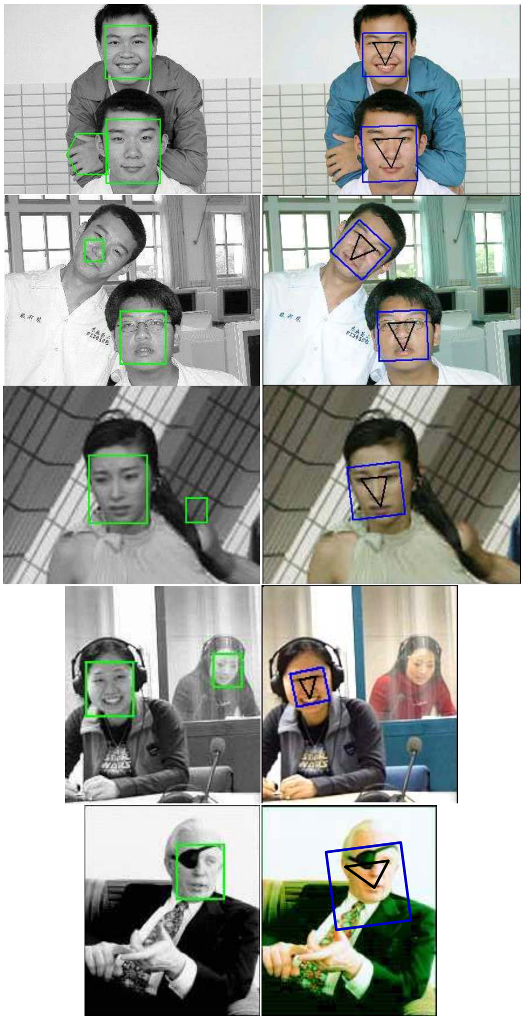


Fig. 9. Comparison of the detection results for CMU face detector and the proposed face detector: the left column is the results of CMU detector and the right column is the results of our detector.

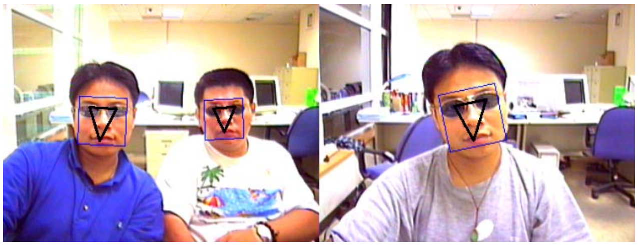


Fig. 10. Results of detecting faces wearing sunglasses.

partitioned into two separated regions and thus causes the wrong detection results. For many common applications, the camera is usually put at a distance from the users. If the distance is far enough, the frames of sunglasses are almost removable after applying the morphological closing operator on the result of skin-color region extraction.

5. Concluding remarks

According to the experimental results, the proposed algorithm exhibits satisfactory performances in both accuracy and speed. Actually, for those applications with well-constrained conditions in system usage and environment control, the proposed algorithm can be further improved in both speed and accuracy by further simplifications and refinement of our system design. However, there still are two main restrictions in using the proposed algorithm:

1. The light condition must be normal. In other words, the faces to be detected cannot be too bright or too dark. In addition, the proposed algorithm does not allow the vast shadows on the faces because they might interfere with the geometry properties of facial components.
2. The facial components must appear on the images as clearly as possible. Thus, our algorithm cannot detect those incomplete faces resulted from serious occlusions and large orientations.

In the future, we plan to improve the algorithms in two directions:

1. increasing the robustness of light variations by developing a light compensation/correction pre-processing technique; and
2. developing more improved component-based detection and verification process for incomplete facial components.

The first is aiming at loosening the restrictions on light condition and the second is for handling the problems of occlusions and large orientations.

Acknowledgements

This work was supported in part by the National Science Council of Republic of China under Grant NSC-90-2218-E-259-001.

References

[1] Pentland A, Moghaddam B, Stamer T, Oliyide O, Turk M. View-based and modular eigenspaces for face recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, Seattle, 1994. p. 84–91.

[2] Belhumeur PN, Hespanha JP, Kriegman DJ. Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence 1997;19(7):711–20.

[3] Er MJ, Wu SQ, Lu JW, Toh HL. Face recognition with radial basis function (rbf) neural networks. IEEE Transactions on Neural Networks 2002;13(3):697–710.

[4] Kim HC, Kim D, Bang SY. Face recognition using the mixture-of-eigenfaces method. Pattern Recognition Letters 2002;23(13): 1549–58.

[5] Gao YS, Leung MKH. Face recognition using line edge map. IEEE Transactions on Pattern Analysis and Machine Intelligence 2002;24(6):764–79.

[6] Rong D, Su GD, Lin XG. Face recognition algorithm using local and global information. Electronics Letters 2002;38(8):363–4.

[7] Kim KI, Jung K, Kim HJ. Face recognition using kernel principal component analysis. IEEE Signal Processing Letters 2002;9(2): 40–2.

[8] Wu JX, Zhou ZH. Face recognition with one training image per person. Pattern Recognition Letters 2002;23(14):1711–9.

[9] Kim KI, Kim JH, Jung K. Face recognition using support vector machines with local correlation kernels. International Journal of Pattern Recognition and Artificial Intelligence 2002;16(1):97–111.

[10] Frischholz RW, Dieckmann U. Bioid: a multimodal biometric identification system. IEEE Computer 2000;3(2):63–8.

[11] Chellappa R, Wilson CL, Sirohey SA. Human and machine recognition of faces, a survey. Proceedings of the IEEE 1995;83: 705–40.

[12] Rowley HA, Baluja S, Kanade T. Neural network-based face detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 1998;20(1):23–38.

[13] Sung KK. Learning and example selection for object and pattern detection. Phd, MIT Press, Cambridge, MA, 1996.

[14] Colmenarez AJ, Huang TS. Face detection with information-based maximum discrimination. In: IEEE Conference on Computer Vision and Pattern Recognition, 1997 San Juan, Puerto Rico. p. 782–7.

[15] Osuna E, Freund R, Girosi F. Training support vector machines: an application to face detection. In: International Conference on Computer Vision and Pattern Recognition, 1997 San Juan, Puerto Rico. p. 130–6.

[16] Lin SH, Kung SY, Lin LJ. Face recognition/detection by probabilistic decision-based neural network. IEEE Transactions on Neural Networks 1997;8(1):114–32.

[17] Kuchi P, Gabbur P, Bhat PS, Davis S. Human face detection and tracking using skin color modeling and connected component operators. IETE Journal of Research 2002;48(3–4):289–93.

[18] Hsu RL, Adbel-Mottaleb M, Jain AK. Face detection in color images. IEEE Transactions on Pattern Analysis and Machine Intelligence 2002;24(5):696–706.

[19] Soriano M, Martinkauppi B, Huovinen S, Laaksonen M. Adaptive skin color modeling using the skin locus for selecting training pixels. Pattern Recognition 2003;36(3):681–90.

[20] Greenspan H, Goldberger J, Eshet I. Mixture model for face-color modeling and segmentation. Pattern Recognition Letters 2001;22(14):1525–36.

[21] Cho KM, Jang JH, Hong KS. Adaptive skin-color filter. Pattern Recognition 2001;34(5):1067–73.

[22] Yao HX, Gao W. Face detection and location based on skin chrominance and lip chrominance transformation from color images. Pattern Recognition 2001;34(8):1555–64.

[23] Wang YJ, Yuan BZ. A novel approach for human face detection from color images under complex background. Pattern Recognition 2001;34(10):1983–92.

[24] Wang YJ, Yuan BZ. Segmentation method for face detection in complex background. Electronics Letters 2000;36(3):213–4.

[25] Cai J, Goshtasby A. Detecting human faces in color images. Image and Vision Computing 1999;18(1):63–75.

[26] McKenna SJ, Gong SG, Raja Y. Modelling facial colour and identity with gaussian mixtures. Pattern Recognition 1998;31(12): 1883–92.

[27] Chen C, Chiang SP. Detection of human faces in colour images. IEEE Proceedings—Vision Image and Signal Processing 1997;144(6):384–8.

[28] Storring M, Andersen H, Granum E. Estimation of the illuminant colour from human skin colour. In: International Conference on Face and Gesture Recognition, Grenoble, France, 2000. p. 64–9.

[29] Terrillon J, Shirazi M, Fukamachi H, Akamatsu S. Comparative performance of different skin chrominance models and chrominance spaces for the automatic detection of human faces in color images. In: International Conference on Face and Gesture Recognition, 2000 Grenoble, France. p. 54–61.

[30] Ishii H, Fukumi M, Akamatsu N. Face detection based on skin color information in visual scenes by neural networks. In: International Conference on System, Man and Cybernetics, Vol. 5. 1999 Nashville, Tennessee, USA. p. 557–63.

[31] Kawato S, Ohya J. Automatic skin-color distribution extraction for face detection and tracking. In: International Conference on Signal Processing, Vol. 2. 2000 Phoenix, Arizona, USA. p. 1415–8.

[32] Zhang HM, Zhao DB, Gao W, Chen XL. Combining skin color model and neural network for rotation invariant face detection, Advances in Multimodal Interfaces—ICMI 2000, Proceedings, Vol. 1948, 2000 Beijing, China. p. 237–44.

[33] Soriano M, Huovinen S, Martinkauppi B, Laaksonen M. Using the skin locus to cope with changing illumination conditions in color-based face tracking. In: IEEE Nordic Signal Processing Symposium, Kolmarden, Sweden, 2000. p. 383–6.

[34] Jeng S, Liao HYM, Liu Y, Chern M. An efficient approach for facial feature detection using geometrical face model. In: International Conference on Pattern Recognition, 1996 Vienna, Austria. p. 1739–55.

[35] Lin C, Fan K. Human face detection using triangle relationship. In: International Conference on Pattern Recognition, Vol. 2. 2000 Barcelona, Spain. p. 941–5.

[36] Yang J, Waibel A. A real-time face tracker. In: Third IEEE Workshop on Applications of Computer Vision, Sarasota, Florida, 1996. p. 142–7.

[37] Schneiderman H, Kanade T. A statistical method for 3d object detection applied to faces and cars. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 1. 2000 Hilton Head Island, South Carolina, USA. p. 746–51.