A SIMPLE AND ACCURATE COLOR FACE DETECTION ALGORITHM IN COMPLEX BACKGROUND

Yu-Ting Pai, Shanq-Jang Ruan†, Mon-Chau Shie, Yi-Chi Liu

Low Power Systems Lab, Department of Electronic Engineering
National Taiwan University of Science and Technology
No.43, Sec.4, Keelung Rd., Taipei, 106, Taiwan, R.O.C.
†E-mail: sjruan@mail.ntust.edu.tw

ABSTRACT

Human face detection plays an important role in many applications such as video surveillance, face recognition, and face image database management. This paper describes a fast face detection algorithm with accurate results. We use lighting compensation to improve the performance of color-based scheme, and reduce the computation complexity of feature-based scheme. Our method is effective on facial variations such as dark/bright vision, close eyes, open moth, a half-profile face, and pseudo faces. It is worth stressing that our algorithm can also discriminate cartoon and human face correctly. The experimental results show that our approach can detect a frame in 111 msecs with the 92.3% detection rate.

1. INTRODUCTION

Recently, many researches detect face by combining feature- and color-based methods to obtain a high performance and high speed results [1][2]. The advantages of these methods are fast, high detection ratio, and can handle faces with complex background. Therefore, we focus on the color- and feature-based face detection. However, these typical methods have some defects. First, Color-based method is hard to detect the skin-color under different lighting conditions [3][4]. Second, Many researchers find the feature of eyes by detecting eyeball, the white of the eye, or the pupil of the eye. It will result in false detection when the human close eyes or wearing glasses. Third, Most of the traditional algorithms can not discriminate cartoon face and real human face at the same scope. Fourth, The feature-based detection has large computation and operates slowly.

In this paper, we propose a face detection algorithm that can detect human face under the different lighting conditions with high speed and high detection ratio. Moreover, our method gives good performance if face with complex backgrounds, cartoon/human face discrimination, half-profile face, and some facial variations. By applying color-based technique, we separate the skin regions and non-skin regions in YC_bC_r color space. We adopt lighting compensation technique and non-linear color transformation to solve the problem of different lighting conditions. By applying feature-based technique, we extract the facial features according to human eyes, mouth, and height to width ratio of face. The experimental results show that the speed of our approach in the real-time testing is about 111 msecs per frame detection. Besides, after applying to 700 real faces the detection ratio of proposed method is 92.3%.

2. IMPLEMENTATION OF PROPOSED METHOD

Due to color- and feature-based detections can fast and accurately find human, many researchers combine these two methods to obtain real human face in a picture. However, the traditional color-based method is hard to detect the skin-color for the case of different lighting condition, and the typical feature-based method has high computation complexity. In this section, we propose a new lighting compensation scheme to overcome the problem of color-based method and simplify the feature-based detection. A system overview of our face detection algorithm is illustrated in Fig.2, and the details are explained as follows.

2.1. Color space transformation and lighting compensation

In order to apply to the real-time system, we adopt skin-color detection as the first step of face detection. Due to YC_bC_r color space transform is faster than other approaches [5][6], we select this transform to detect human skin. However, the luminance of every image is different. It results that every image has different color distribution. Therefore, our lighting compensation is based on luminance to modulate the range of skin-color distribution. First, we compute the average luminance Y_{aveg} of input image.

$$Y_{aveg} = \sum Y_{i,j} \tag{1}$$

where $Y_{i,j} = 0.3R + 0.6G + 0.1B$, $Y_{i,j}$ is normalized to the range (0,255), and i, j are the index of pixel. According

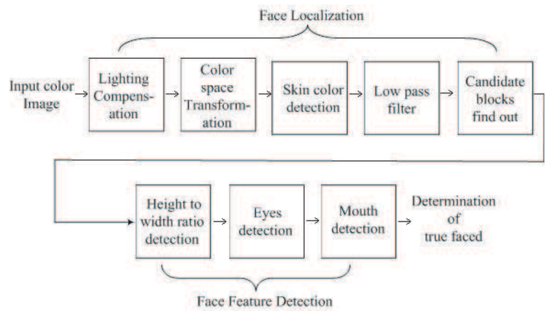
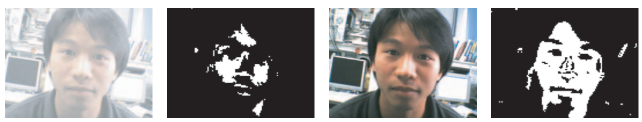


Fig. 1. The flow of proposed mehtod.



(a) (b) (c) (d)

Fig. 2. An example of bright image compensation: (a) original bright image (b) the S_{ij} without compensation (c) compensated image (d) the S_{ij} of compensated image.

to Y_{aveg}, we can determine the compensated image C_{ij} by following equations:

$$R'_{ij} = (R_{ij})^{\tau} \tag{2}$$
$$G'_{ij} = (G_{ij})^{\tau} \tag{3}$$
$$C_{ij} = \{R'_{ij}, G'_{ij}, B_{ij}\}, \tag{4}$$

where

$$\tau = \begin{cases} 1.4, & Y_{aveg} < 64 \\ 0.6, & Y_{aveg} > 192 \\ 1, & \text{otherwise.} \end{cases}$$

Note that we only compensate the color of R and G to reduce computation.

Due to chrominance (C_r) can represent human skin well, we only consider C_r factor for color space transform to reduce the computation. C_r is defined as follow:

$$C_r = 0.5R' - 0.419G' - 0.081B \tag{5}$$

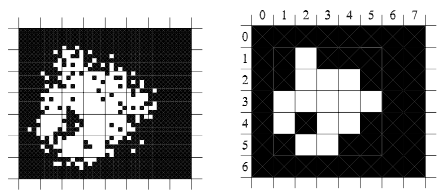
In Eq. (5) we can see that R' and G' are important factors due to their high weight. Thus, we only compensate R and G to reduce computation. According to C_r and experimental experience, we define the human skin by a binary matrix:

$$S_{ij} = \begin{cases} 0, & 10 < C_r < 45 \\ 1, & \text{otherwise} \end{cases} \tag{6}$$

where "0" is the white point, and "1" is black point. Fig. 2 and Fig. 3 show the compensation effect on bright and dark image respectively. We can see that result shown in Fig. 2(d) and Fig. 3(d) are good enough for skin-color detection.

(a) (b) (c) (d)

Fig. 3. An example of dark image compensation: (a) original dark image (b) the S_{ij} without compensation (c) compensated image (d) the S_{ij} of compensated image.



(a) (b)

Fig. 4. (a)An example of S_{ij} (b)remove noise by the 5×5 low pass filtern.

2.2. High frequency noisy removing

In order to remove high frequency noise fast, we implement a low pass filter by a 5×5 mask. First, we segment S_{ij} into 5×5 blocks, and calculate how many white points in a block. Then, every point of a 5×5 block is set to white point when the number of white points is greater than half number of total points. On the other hand, if the number of black points is more than a half, this 5×5 block is modified to a complete black block. Fig. 4(b) shows an example that we remove high frequency noise from Fig. 4(a). Although this fast filter will bring block effect, it can be disregarded due to that our target is to find where is human skin.

2.3. Find out the skin-color blocks

After performing the low pass filter, there are several skin-color regions may be human face will be in $S_{i,j}$. In order to mark these region, we store fore vertices of rectangle for every region. First, we find the leftmost, rightmost, upmost, and downmost points. By these four points, we create a rectangle around this region. Fig. 4(b) shows an example that store (1,1), (1,5), (5,1), and (5,5) to describe the candidate region. Thus, we can get several skin-color blocks called candidate blocks to detect facial feature.

2.4. Height to width ratio detection

After the step of face localization, we can get several regions which may be human face. Then, the feature of height to width ratio, mouth, and eyes are detected sequentially for every candidate block. Because any of these three detections can reject the candidate blocks, low computation module has high priority to process. Height to width ratio is a very fast

(a) (b) (c)

Fig. 5. An example of mouth detection. (a) original candidate block (b) mouth pixels detection vision (c) M_{hw} based vertical histogram.

(a) (b) (c)

Fig. 6. An example of eyes detection. (a) original candidate block (b) eyes pixels detection vision (c) $E_{\hat{h}w}$ based vertical histogram.

and simple detection. Let the size of candidate block is $h \times w$. We define that if the height to width ratio ($h : w$) is out of range between 1.5 and 0.8, it should be not a face and this candidate block will be discarded. Note that the range is determined by experiments. If the ratio is between 1.5 and 0.8 may be a face, the block should be processed by the following two detections.

2.5. Mouth detection

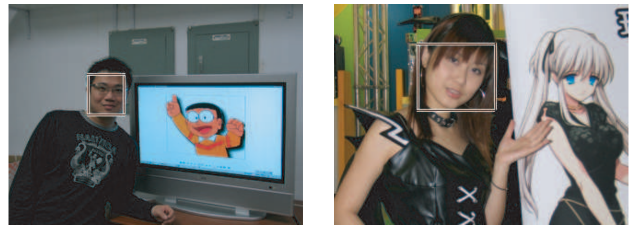
After determining the height to width ratio for the candidate blocks, a more complex detection will be applied to find mouth feature. We use θ proposed by [7] to find the mouth pixels. The θ value is calculated for all of the pixels in every candidate block. The θ is defined as:

$$\theta = \cos^{-1}\left(\frac{0.5(2R' - G' - B)}{\sqrt{(R' - G')^2 + (R' - B)(G' - B)}}\right). \qquad (7)$$

The pixel will be determined to be part of mouth by a binary matrix M:

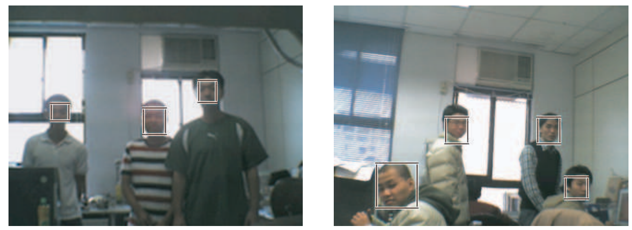
$$M_{pq} = \begin{cases} 0, & \theta < 90 \\ 1, & \text{otherwise,} \end{cases} \qquad (8)$$

where "0" means that pixel is mouth. Fig. 5(a) and (b) is an example for mouth pixel detection. In Fig. 5 (b), the mouth pixel is presented by white point. Then, we use vertical based histogram to determine whether or not it is a mouth in this block. We calculate how many mouth pixels are in the same



(a) (b)

Fig. 7. Face detection results on cartoon personage and real human.



(a) (b)

Fig. 8. Face detection results on (a) dark and (b) bright light vision.

y-coordinates, and use w_h to store the value of different y-coordinates. Fig. 5(c) illustrates an example of the histogram of Fig. 5(b). Note that the maximum value of w_h is denoted by w_{max}, and the y-coordinate of w_{max} is represented by h_m. Thus, we define if w_{max} is less than $1/6$ block width w, this block will be rejected. For example, in Fig. 5(c) w_{max} is more than $(1/3)w$, we can know that the mouth feature is embedded in this block.

2.6. Eyes detection

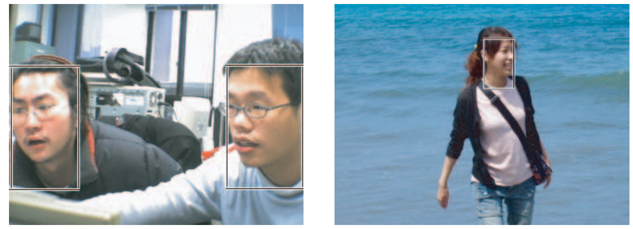
After mouth detection stage, we know that the y-coordinate of mouth is h_m and the y-coordinate of eyes must smaller than h_m according to our definitions. This information let us to detect human eyes in the smaller region. The region is defined by the y-coordinate 0 to $h_m - w_{may}$. Because the y-coordinate of mouth is must larger than eyes, the considered height of region must be less than h_m. An example of detecting region is shown in Fig. 6(a). Due to the deeper lineaments around human eyes, we can detect the existence of human eyes by the luminance which is slightly darker than average skin-color. The pixels which around human eyes is defined by $E_{\hat{h}w}$:

$$E_{\hat{h}w} = \begin{cases} 0, & 65 < Y < 80 \\ 1, & \text{otherwise,} \end{cases} \qquad (9)$$

where $\hat{h} = h_m - w_{max}$. Fig. 6(b) shows an example that we find out the pixels around eyes. Then, the vertical based histogram, illustrated in Fig. 6(c), shows the distribution of $E_{\hat{h}w}$. In this histogram, we assume the candidate block has human

(a) (b)

Fig. 9. Face detection results on facial variations: (a) closed eyes with wearing glasses (b) open mouth.



(a) (b)

Fig. 10. Face detection results on half-profile faces.

eyes if there exist a α value greater than a threshold β. Here we let $\alpha = 0.5w_{max}$ and $\beta = w_{max}$. When we finish the eyes detection, we regard the blocks which pass three feature detections are human face.

3. EXPERIMENTAL RESULTS

We evaluated proposed algorithm by 300 still images with 700 true human faces and 20 pseudo faces. These test images were made by 12 brands of cameras which include Canon, Nikon, Sony, Fujifilm, and Olympus etc. The size of image is 320×240 pixels. The pseudo face is cartoon personage. The experimental result shows that our algorithm can identify 646 real faces in our test image. It means that the detection rate is 92.3%. We only miss 54 faces and 47 face detection errors. Fig. 7 shows a successful example that we identify the real face and reject the cartoon personage. Fig. 8(a) and Fig. 8(b) demonstrate that our proposed algorithm can successfully identify dark and bright skin faces respectively. Fig. 9 shows that our algorithm can successful detect face under some facial variations such as closed eyes, open mouth, and wearing glasses. Furthermore, Fig. 10 demonstrates that our algorithm also can detect nonfontal face successfully. Obviously, our method gives good performance with respect to complex backgrounds, different light sources, some facial variations, and half-profile face. Besides, the processing speed in real-time testing is 111 msecs for a frame. Thereby our proposed algorithm is suitable for use in real-time systems with high performance.

4. CONCLUSION

In this paper, we proposed an accurate and high speed color face detection system for complex background. The color- and feature-based detections were adopted to find skin-color fast and selected candidate blocks carefully. We used lighting compensation to improve the performance of color-based scheme, and reduce the computation of feature-based scheme. The major contribution of this paper is that the proposed method can detect successfully under dark/bright vision, close eyes, open mouth, wearing glasses, and half-profile face. It is worth stressing that our algorithm can discriminate cartoon and human face correctly. With the high detection rate 92.3%, our approach can quickly detect a frame in 111 msecs. In future work, the proposed approach can be applied to hardware implementation. Due to the proposed method has simple structure, it is suitable to be implemented in hardware to achieve very high performance and low power system.

5. REFERENCES

[1] Q. Yuan, W. Gao, and H. Yao, "Robust frontal face detection in complex environment," in *16th International Conference on Pattern Recognition, 2002. Proceedings*, Aug. 2002, vol. 1, pp. 25–28.

[2] R.-L. Hsu, M. Abdel-Mottaleb, and A. K. Jain, "Face detection in color images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 696–706, May 2002.

[3] J. Kovac, P. Peer, and F. Solina, "Illumination independent color-based face detection," in *Proceedings of the 3rd International Symposium on Image and Signal Processing and Analysis*, Sept. 2003, vol. 1, pp. 510–515.

[4] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 23–38, Jan 1998.

[5] S. Gundimada, Li Tao, and v. Asari, "Face detection technique based on intensity and skin color distribution," in *2004 International Conference on Image Processing*, Otc. 2004, vol. 2, pp. 1413–1416.

[6] K. P. Seng, A. Suwandy, and L.-M. Ang, "Improved automatic face detection technique in color images," in *IEEE Region 10 Conference TENCON 2004*, Nov. 2004, vol. 1, pp. 459–462.

[7] Y. Araki, N. Shimada, and Y. Y. Shirai, "Detection of faces of various directions in complex backgrounds," in *16th International Conference on Pattern Recognition, 2002. Proceedings*, Object 2002, vol. 1, pp. 409–412.