Face detection using SVM-based classification

Article · January 2009

Far East Journal of Experimental and Theoretical Artificial Intelligence
Volume 3, Issue 2, 2009, Pages 113-123
This paper is available online at http://www.pphmj.com

FACE DETECTION USING SVM-BASED CLASSIFICATION

JIH PIN YEH*, I-CHUN PAI, CHUN-WEI WANG, FU-WEN YANG and HWEI-JEN LIN

*Department of Information Management System
Chinmin Institute of Technology
Miaoli, Taiwan

Department of Computer Science and Information Engineering
Tamkang University
Taipei, Taiwan

Abstract

This paper proposes an improved version of our previously introduced face detection system based on skin color segmentation and neural networks. The new system, using a support vector machine (SVM) based method for learning and verification, consists of several stages. First, the system searches for the regions where faces might exist by using skin color information and forms a so-called skin map. After performing noise removal and some morphological operations on the skin map, it utilizes the aspect ratio of a face to find out possible face blocks, and then eye detection is carried out within each possible face block. If an eye pair is detected in a possible face block, a region is cropped according to the location of the two eyes, which is called a face candidate; otherwise, it is regarded as a non-face block. Finally, each of the face candidates is verified by a support vector machine. Experimental results reflect that the new version improves the verification accuracy of the previously proposed system.

Keywords and phrases: face detection, skin color segmentation, RGB color space, HSV color space, support vector machine (SVM).

Received June 24, 2009

I. Introduction

Recently, face detection has received much attention and has been an extensive research topic. It is the important first step of many applications such as the face recognition system proposed by Kondo and Yan [6], facial expression analysis, surveillance systems, the work on video-conferencing done by Shi et al. [10], intelligent human-computer interaction, content-based image retrieval systems, etc. Therefore, the effectiveness of face detection influences the performance of these systems. There have been various approaches proposed for face detection, which could be generally classified into four categories: (i) Template matching methods, (ii) Feature-based methods, (iii) Knowledge-based methods, and (iv) Machine learning methods. Template matching method means the final decision comes from the similarity between input image and template. It is scale-dependent, rotation-dependent and computationally expensive. Feris et al. [2] verified the presence or absence of a face in each skin region by using an eye detector based on a template matching scheme. Suzuki and Shibata [12] utilized edge distribution of face images and non-face images to generate several 64-dimensional feature vectors which could be regarded as templates. Then the feature vector of the input image is compared with all the templates and classified as face or non-face depending on the matching result. Wang and Tan [16] presented a deformable template based on the edge information to match the face contour. Feature-based methods adopt low-level features to find out the face location such as illumination variation [4, 17], color information [7, 16], edge information [6], and shape feature [16]. Recently, face detection algorithms based on skin color information has attracted more attention of many researches. Therefore, the accuracy of skin color detection is very important to a face detection system. Wong et al. [18] proposed an efficient color compensation scheme for skin color segmentation. Cho et al. [1] proposed an adaptive skin color filter for detecting skin color regions in a color image. However, it failed to detect the skin color regions when an input image is composed of several different races. Vezhnevets et al. [15] surveyed numerous pixel-based skin color detection techniques. Lin and Fan [8] proposed a nonedge-based method to detect an isosceles triangle (for frontal view) or a right triangle (for side view). Machine learning methods use a lot of training samples to make the machine to be capable of judging face or non-face. Turk and Pentland [13] presented eigenfaces for face recognition. Karungaru et al. [5] used a fixed window to scan whole image, then the input image is verified by a back-propagation neural network.

We previously proposed an approach for detecting human faces in color images under different illumination conditions, scale, rotation, with/without glasses, based on classification by neural networks. In this paper, we modify the approach by using SVM for classification. First, skin color segmentation is performed to find skin color regions. Secondly, possible face blocks are located by using some restrictions on these regions. Thirdly, eye detection and matching are carried out within each possible face block, and then face candidates will be obtained according to the locations of the detected eye pairs. Finally, each of the face candidates is verified by a SVM classifier.

The rest of this paper is organized as follows: In Section II, we describe how to obtain the face candidates. The process of face candidate verification based on SVM is presented in Section III. Section IV provides some experimental results and comparison with our previous method. Finally, conclusions are given in Section V.

II. Face Candidate Searching

The main purpose of this section is to obtain face candidates. There are three steps to achieve this goal. First, skin color regions are located by performing skin color segmentation. Second, some restrictions to these regions are used to locate possible face blocks. Finally, eye detection and matching are implemented within each possible face block, and then face candidates are determined according to the locations of the detected eye pairs.

2.1. Skin color segmentation

Skin color is a very important feature of human faces. The distribution of skin colors clusters in a small region of the chromatic color space [7]. Processing color is faster than processing other facial features. Therefore, skin color detection is first performed on the input color image to reduce the computational complexity. Because of the accuracy of skin color detection affects the result of face detection system, choosing a suitable color space for skin color detection is very important. Among numerous color spaces, RGB color space is sensitive to the variation of intensity, and thus it is not sufficient to use only RGB color space to detect skin color.

In this paper, we combine RGB, normalized RGB, and HSV color spaces to detect skin pixels. This is due to the fact that both normalized RGB and HSV color space can reduce the effect of lighting to an image. In the normalized RGB color space, it suffices to represent color using only two values, say (r, g), as defined in (1), since the blue component can be obtained from $r + g + b = 1$. The HSV color space can be converted from the RGB color space using (2),

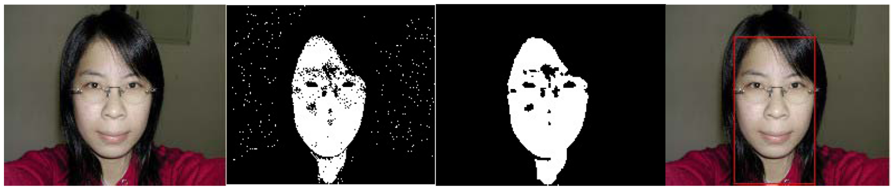
$$r = R/(R + G + B), \qquad g = G/(R + G + B), \tag{1}$$

$$H1 = \cos^{-1}\left\{\frac{0.5[(R - G) + (R - B)]}{\sqrt{(R - G)^2 + (R - B)(G - B)}}\right\},$$

$$H = \begin{cases} HI & \text{if} \quad B \leq G, \\ 360^\circ - HI & \text{if} \quad B > G. \end{cases}$$

$$S = \frac{\mathrm{Max}(R, G, B) - \mathrm{Min}(R, G, B)}{\mathrm{Max}(R, G, B)}, \quad V = \frac{\mathrm{Max}(R, G, B)}{255}. \tag{2}$$

At first, RGB values of pixels in the input image are transformed into the normalized RGB color space and the HSV color space. A pixel is labeled as a skin pixel if its color values conform to conditions given in (3) to (6). As a result, we can generate a binary skin map where the white points represent the skin pixels and the black points represent the non-skin pixels. Then, we apply median filter to the skin map for noise removing and perform morphological opening operation proposed with structuring element of size 3×3 to eliminate small skin blocks. Afterwards, utilizing connected component operation to find out all connected skin regions and each of the skin regions is labeled by a bounding box. The process is depicted in Figure 1.



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

Figure 1. (a) Original image, (b) Skin map, (c) Skin map performed median filter followed by opening and (d) Result of skin region detection.

$$R > G, \quad |R - G| \geq 11, \tag{3}$$

$$0.33 \leq r \leq 0.6, \quad 0.25 \leq g \leq 0.37, \tag{4}$$

$$340 \leq H \leq 359 \vee 0 \leq H \leq 50, \tag{5}$$

$$0.12 \leq S \leq 0.7, \quad 0.3 \leq V \leq 1.0. \tag{6}$$

2.2. Locating possible face blocks

To check if a skin region contains a face, we use three constraints: *Area size*, *Aspect Ratio*, and *Occupancy*. Each skin region satisfying the three constraints is considered as a possible face block; otherwise, a non-face region.

The first constraint is that the size of the bounding box surrounding the skin region, denoted by *Area size*, is greater than 30×30. This is due to the fact that in a skin region of too small size, facial features might be eliminated during the pre-processing.

The second constraint is that the ratio of the height to the width of the bounding box, denoted by *Aspect Ratio*, is between 0.8 and 2.6. This rule is based on observations on various face blocks.

Occupancy denotes the ratio of the amount of the skin pixels to the size of the bounding box. If the bounding box contains a face, the number of skin pixels in the bounding box must be large enough. We relax the restriction to avoid removing a real face and set the third constraint as "*Occupancy* is greater or equal to 40%".

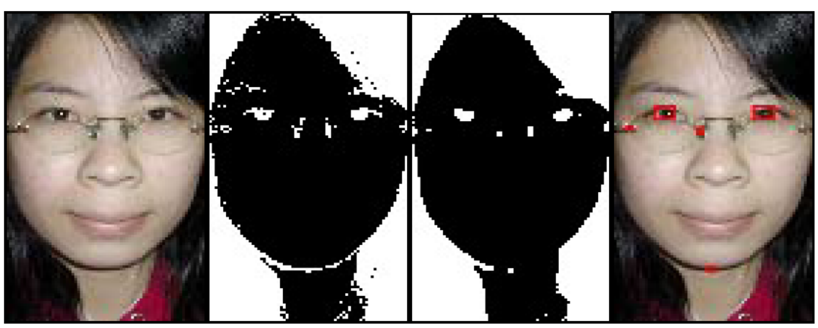
The skin regions that do not obey all the three constraints are removed, and the remaining are considered as possible face blocks and need further verification.

2.3. Determining face candidates

Eye is the most stable facial feature of human face. Suppose face is in frontal view, there must be two eyes in the possible face block. Hence, eye detection and matching is carried out to detect any eye pair existing in the possible face blocks. During the procedure, a possible face block is removed if it contains no eye or only one eye; otherwise, according to the location of the detected eye pair, a face candidate is located.

2.3.1. Eye detection

In general, the intensity of eye is darker than that of other facial features in a face and it does not belong to skin region. Utilizing this property of eyes, we can find out some eye pixels and present them with white points in the possible face block, as shown in Figure 2(b). Since some noises are simultaneously produced during this process, we apply median filter to remove them (as shown in Figure 2(c)) and then perform connected component operation to find all eye-like blocks. Each of the eye-like blocks is labeled by a bounding box, and then examined by three conditions to verify if it contains an eye. The first condition is that the Aspect Ratio of the eye-like block must be between 0.2 and 1.67. The second condition is that the Occupancy must be greater than or equal to 30%. The third condition is that the ratio of the width of the eye-like block to the width of the possible face block is between 0.028 and 0.4. These parameters came from experimental results. Examining these conditions, eye blocks can be detected, as shown in Figure 2(d).



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

Figure 2. (a) Possible face block, (b) Eye pixels, (c) (b) Eye pixels after performing median filter and (d) Result of eye detection.

2.3.2. Eye matching

Over the detected eye blocks, we find eye pairs for frontal view based on the geometrical relation of facial features. For each pair of eye blocks, we first locate the centroid of each of the two eye blocks and calculate the horizontal distance, *Dist*, between two centroids. Second, we match the two eye blocks based on the following rules: "*Dist*, is between T_1 and T_2 times of the width of the face," "The eye is located at upper portion of the face," and "The sizes of the two eyes are nearly equal." In our experiments, we set $T_1 = 0.2$, $T_2 = 0.65$. As soon as an eye pair is located, we can clip a face candidate based on the face model as shown in Figure 3, where D is the distance of the centroids of the two eyes.

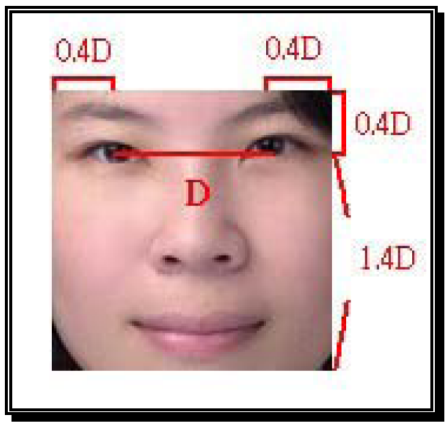


Figure 3. Face model.

III. Face Candidate Verification using SVM

Support Vector Machine (SVM), proposed by Vapnik [14], is a new machine learning technology that has been widely and successfully applied in solving many kinds of pattern recognition problems. The performance of this technique is better than traditional supervised machine learning approaches, including neural networks, *K*-nearest neighbor classifiers. For example, it can avoid the overfitting problems of neural networks. The basic work of SVM is to transform the samples into a high-dimensional Hilbert space and to compute a hyperplane that best separates the samples, as described by Stitson et al. [11].

The previous section shows how to clip the face candidates according to the detected eye pairs. In this section, a SVM based classifier is used to verify each face candidate. The SVM used the radial basis function (RBF) kernel as shown in (7), where we set $\gamma = 20$ for our experiments. Before training, each of the training samples is intensity-normalized by utilizing (8) and (9), where N is the number of pixels in a training sample, $\bar{I}$ is the average gray value, I_i is the gray value of the ith pixel, and I_i' is the normalized gray value

$$K(x_i, x_j) = \exp(-\gamma(x_i - x_j)^2), \tag{7}$$

$$\bar{I} = \frac{1}{N}\sum_{i=1}^{N} I_i, \tag{8}$$

$$I_i' = (I_i - \bar{I}) + 128. \tag{9}$$

After training by SVM, a face candidate can be verified through the following steps:

(1) Normalize the size of the face candidate to size of 20×20 by the Nearest Neighbor method.

(2) Apply intensity-normalization to the face candidate by using equations (8) and (9).

(3) Input face candidate into the SVM classifier to produce the output.

IV. Experimental Results

In this section, we show a set of experimental results to demonstrate the performance of the proposed system. Our experimental environment is a personal computer with an AMD Athlon 2200+ 1.8GHz CPU, 512MB of memory, and Windows XP.

We tested our system on 1817 color images, taken from digital cameras, scanners, the World Wide Web, and the Champion dataset. They consist of both indoor and outdoor scenes under different lighting conditions and backgrounds. The image sizes vary from 105×158 to

640×577. In these images, there are totally 2615 faces of sizes varying from 35×30 to 370×275 pixels. Faces vary in lighting, scale, position, rotation, race, color, and facial expression. The average time required for our system to detect faces in one frame is about 85 ms.

In the experiments, we compare the proposed method with our previous method based on classification by neural networks. As the comparison given in Table 1, the method proposed by Fröba and Küblbeck [3] and our previously and presently proposed methods have false acceptance rates (FAR) 12.1%, 11.7%, and 2.8%, respectively, and have false rejection rates (FRR) 13.5%, 5.8%, and 1.6%, respectively. Figure 4 shows some face detection results of our system.

Figure 4. Some face detection results of the proposed method.

Table 1. Comparison of three methods

Results / Methods	Test images include 2615 faces	
	FRR	FAR
Method proposed by Fröba et al.	12.1%	13.5%
Previously proposed method	11.7%	5.8%
Presently proposed method	2.8%	1.6%

V. Discussion and Conclusions

This paper proposes a human face detection system based on skin color segmentation and SVM classification. Experimental results show that the proposed system results in better performance than our previously proposed method, proposed by Lin et al. [9], based on neural network classification, which has been proved to outperform the method proposed by Fröba et al., in terms of correct detection rate and capacity of coping with the problems of lighting, scaling, rotation, and multiple faces.

Although the proposed method shows high detection rate, it still has some problems as stated in the following:

(1) The use of skin color information causes the system to fail in skin color detection when the illumination is too bright or too dark.

(2) Since we determine a face candidate according to the location of two eyes, when eyes are not successfully detected, the system would fail in face detection.

In our future work, we would like to solve these problems. For the first problem, we will try to find a more robust skin color detection algorithm. For the second problem, we might try to adequately relax the rules for eye pair detection.

References

[1] K. M. Cho, J. H. Jang and K. S. Hong, Adaptive skin-color filter, Pattern Recognition 34(5) (2001), 1067-1073.

[2] R. S. Feris, T. E. Campos and R. M. Cesar, Jr., Detection and tracking of facial features in video sequences, Lecture Notes on Artificial Intelligence, Vol. 1793, 2000, pp. 127-135.

[3] B. Fröba and C. Küblbeck, Face Detection and Tracking Using Edge Orientation Information, SPIE Visual Communications and Image Processing, 2001, pp. 583-594.

[4] C. C. Han, H. Y. Mark Liao, G. J. Yu and L. H. Chen, Fast face detection via morphology-based pre-processing, Pattern Recognition 33(10) (2000), 1701-1712.

[5] S. Karungaru, M. Fukumi and N. Akamatsu, Human face detection in visual scenes using neural networks, Transaction of Institute of Electrical Engineers of Japan (IEEJ) 122c (2002), 995-1000.

[6] T. Kondo and H. Yan, Automatic human face detection and recognition under non-uniform illumination, Pattern Recognition 32 (1999), 1707-1718.

[7] J. Kovaè, P. Peer and F. Solina, Human skin color clustering for face detection, International Conference Proc.-Computer as a Tool (EUROCON 2003) 2, 2003, pp. 144-148.

[8] C. H. Lin and K. C. Fan, Triangle-based approach to the detection of human face, Pattern Recognition 34(6) (2001), 1271-1284.

[9] H. J. Lin, S. Y. Wang, S. H. Yen and Y. T. Kao, Face detection based on skin color segmentation and neural network, International Proc.-Neural Networks and Brain (ICNN&B'05), Beijing, China, 2005, pp. 1144-1149.

[10] Cuizhu Shi, Keman Yu, Jiang Li and Shipeng Li, Automatic image quality improvement for videoconferencing, IEEE Proc.-Acoustics, Speech and Signal Processing (ICASSP'04), Vol. 3, 2004, pp. 701-704.

[11] M. O. Stitson, J. A. E. Wetson, A. Gammerman, V. Vovk and V. Vapnik, Theory of Support Vector Machines, Technical Report CSD-TR-96-17, Royal Holloway University of London, England, 1996.

[12] Y. Suzuki and T. Shibata, Multiple-clue face detection algorithm using edge-based feature vectors, IEEE Proc.-Acoustics, Speech and Signal Processing (5) V (2004), 737-740.

[13] M. Turk and A. Pentland, Eigenfaces for recognition, J. Cognitive Neuroscience 3(1) (1991), 71-86.

[14] V. N. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag, New York, 1995.

[15] V. Vezhnevets, V. Sazonov and A. Andreeva, A survey on pixel-based skin color detection techniques, Proc.-Graphicon-2003, 2003, pp. 85-92.

[16] J. G. Wang and T. N. Tan, A new face detection method based on shape information, Pattern Recognition Letters 21 (2000), 463-471.

[17] K. W. Wong, K. M. Lam and W. C. Siu, An efficient algorithm for human face detection and facial feature extraction under different conditions, Pattern Recognition 34 (2001), 1993-2004.

[18] K. W. Wong, K. M. Lam and W. C. Siu, An efficient color compensation scheme for skin color segmentation, IEEE Proc.-Circuits and Systems (ISCAS2003), Vol. II, 2003, pp. 676-679.