Face Detection in Non-uniform Illumination Conditions by Using Color and Triangle-based Approach

Chiunhsiun Lin

National Taipei University, Taipei, Taiwan, 10433

Abstract

This investigation develops an efficient face detection scheme that can detect multiple faces in color images with complex environments and non-uniform illumination conditions. The proposed scheme comprises two stages. The first stage adopts YES color and triangle-based segmentation to search potential face regions. The second stage involves face verification using a weighting mask function. The system can handle various sizes of faces, non-uniform illumination conditions, pose and changeable expression. In particular, the scheme significantly increases the execution speed of the face detection algorithm in complex backgrounds. A result of this study performs better than previous methods in terms of speed and ability to handle non-uniform illumination conditions.

Keywords: Color segmentation; Face detection; Triangle-based Approach; Weighting mask function.

1. Introduction

Abundant of researches have been conducted on human face detection, for an excellent survey can be found in [1~2]. Some important researches can see [3~9]. In this paper, a robust and efficient human face detection system is proposed to use this valuable component — color. Our approach can appreciably increase than previous methods. The proposed system is composed of two principal parts as shown in Fig. 1.

The rest of the paper is organized as follows. In section 2, segmentation of potential face regions based on skin color segmentation in YES color space and triangle-based approach is described. In section 3, each of the normalized potential face regions is fed to the weighting mask function to verify whether the potential face region really contains a face. Experimental results are demonstrated in section 4 to verify the validity of the proposed face detection system. Finally, conclusions are given in section 5.

2. Searching for Potential Face Regions

After performing lots of tests in various color spaces, including the RGB and HSV spaces, we decided to utilize the YES color space that Y represents the luminance component and E and S denote the chrominance factors. In the YES color space, the luminance component (Y) is a weighted sum of the RGB values, while the chrominance factors are spectral differences: The signal in the E factor is proportional to the difference of the red and green color channels, while the S color factor is proportional to yellow minus blue. The YES color space can be transformed from RGB coordinates. The transformed matrix is given by RGB - YES Conversion Matrix is shown as Fig. 2. The YES space was chosen because: (1) In the YES color space, the luminance component (Y) is independent factor, so we can use this characteristic to conquer the problem of illumination variation. (2) According to Saber and Tekalp [3], the YES color space provides a better representation because of its computational efficiency and absence of nonsingularities in the color transformation process, and the YES color space can reduce variations in chrominance due to changes in luminance. (3) Using the YES color space because of the minimum overlap between skin and non-skin data. Therefore, we collected the human skin colors with different illumination circumstances from 1000 persons, and collected 100 pixels of the human skin colors from each person. Therefore, there are 100,000 pixels of the human skin colors were tested, and we obtained the YES color space distribution. If the pixels of the input image are regarded as skin color, and will keep the original color. Otherwise, we will treat the pixels as non-skin color, and assign them to pure black color. When we read in a color image, we will go on human skin color segmentation task firstly. Our principle is completely different to Cho et al. [4] method that tried to effectively separating skin color regions from seemingly similar, but different color regions. Our principle is that the output of human skin color segmentation must include all the similar skin color (even some of them are seemingly similar or not real human skin color). In other words, the similar light colors should not be missed in the human skin color segmentation process. The output pixels may have other light colors as well because the other light colors will be eliminated during the binarization process. Fig. 3 displays our skin color segmentation scheme is robust to different skin colors. Fig. 3 (a) shows the input image; Fig. 3 (b) illustrates the output result.

Here, we use raster scanning (left-to-right and top-to-bottom) to get 4-connected components, label them, and then find the center of each block. Two pixels p and q with values from V (the set of pixels) are 4-connected if q is in the set N_4 (p). In other words, if a pixel p at coordinates (x, y) has four horizontal and vertical neighbors whose coordinates are (x+1, y), (x-1, y), (x, y+1), (x, y-1). This set of pixels, called the 4-neighbors of p, is denoted by N_4 (p). The detail of raster scanning can be found in the textbook written by Rafael C. et al. [10].

We discover that two eyes and one mouth in the frontal view will form an isosceles triangle. This is the rationale on which the finding of potential face regions is based. If the triangle i j k is an isosceles triangle as shown in Fig. 4(a), then it should possess the characteristic of "the distance of line i j = the distance of line j k". From observation, we discover that the Euclidean distance between two eyes (line i k) is about 90% to 110% of the Euclidean distance between the centers of the right/left eye and the mouth. Due to the imaging effect, imperfect binarization result and various poses of human faces, a 25% deviation is given to absore the tolerance. The first matching rule can thereby be stated as (abs(D(i, j)-D(j, k)) < 0.25*max(D(i, j), D(j, k))), and the second matching rule is (abs(D(i, j)-D(i, k)) < 0.25*max(D(i, j), D(j, k))). Since the labeling process is operated from left to right then from top to bottom, we can get the third matching rule as "i < j < k". Here, "abs" means the absolute value, "D (i, j)" denotes the Euclidean distance between the centers of block i (right eye) and block j (mouth), "D (j, k)" denotes the Euclidean distance between the center of block k (left eye) and block j (mouth), "D (i, k)" represents the Euclidean distance between the centers of block i (right eye) and block k (left eye). For example, as shown in Fig. 4(a), if three points (i, j, and k) satisfy the matching rules, then we think that they construct an isosceles triangle.

After we have found the two eyes and one mouth, we use the Euclidean distance between the centers of block i (right eye) and block j (left eye) to get the coordinates of the four corner points that form the potential facial region as shown in Fig. 4(b). Finally, clip the blocks that satisfy the triangle criteria as the potential face regions as shown in Fig. 4(b). Since we think that the real facial region should cover the eyebrows, two eyes, mouth and some area below the mouth, the coordinates can be calculated as follows:

$$X1 = X4 = Xi - 1/4*D\ (i, k); \qquad (1)$$

$$X2 = X3 = Xk+1/4*D\ (i, k); \qquad (2)$$

$$Y1 = Y2 = Yi+1/4*D\ (i, k); \qquad (3)$$

$$Y3 = Y4 = Yj -1/4*D\ (i, k); \qquad (4)$$

Assume that (X_i, Y_i), (X_j, Y_j) and (X_k, Y_k) are the three center points of blocks i, j, and k, that form an isosceles triangle. (X1, Y1), (X2, Y2), (X3, Y3), and (X4, Y4) are the four corner points of the face region as shown in Fig. 4. X1 and X4 locate at the same coordinate of (X_i - 1/3*D (i, k)); X2 and X3 locate at the same coordinate of (X_k + 1/3*D (i, k)); Y1 and Y2 locate at the same coordinate of (Y_i + 1/3*D (i, k)); Y3 and Y4 locate at the same coordinate of (Y_j - 1/3*D (i, k)); where D (i, k) is the Euclidean distance between the centers of block i (right eye) and block k (left eye).

3. Face Verification

Normalization of a potential face region can reduce the effects of variation in the distance and location. Since all potential faces will be normalized to a standard size (e.g. 60 * 60 pixels) in this step, the potential face regions that we have selected in the previous section are allowed to have different sizes. Here, we resize the potential facial region using ″bicubic″ interpolation technique. The detail of ″bicubic″ interpolation technique can be found in the textbook written by Rafael C. et al. [10].

If the normalized potential facial region really contains a face, it should have high similarity to the mask that is formed by 10 binary training faces. The method for generating a mask is to read in 10 binary training masks that are cut manually from the facial regions of images. Then add the corresponding entries in the 10 training masks to form an added mask. In other words, we use 10 binary faces to structure a template mask that is used for comparing with each normalized potential facial region. The idea is similar to the neural network concept, but it does not need so many training samples. Each normalized potential facial region is fed into the weighting mask function that is used to compute the similarity between the normalized potential facial region and the mask. The computed value can be utilized in deciding whether a potential region contains a face or not.

The algorithm for obtaining the weight of the potential facial region is stated as follows:

Input: the potential facial region and the mask of the training samples

Output: the weight of the potential facial region

```
Begin
For all pixels of the potential facial region:
    Step 1) If the pixel of potential facial region is black and the pixel of the mask is also black, then weight = weight + 6.
    Step 2) If the pixel of potential facial region is white and the pixel of the mask is also white, then weight = weight + 2.
    Step 3) If the pixel of potential facial region is black and the pixel of the mask is white, then weight = weight - 4.
    Step 4) If the pixel of potential facial region is white and the pixel of the mask is black, then weight = weight –2.
    Step 5) Calculate the weight of the potential facial region for all pixels.
End
```

A set of experimental results demonstrates that the threshold values of the human face should be set between 4,000 and 5,500.

4. Experimental Results

There are 1200 test images (include 560 different persons). Some test images are taken from digital camera, some from digital video, some from scanner, and some from videotape. Moreover, we use some parts of the "AR face database" [11] to verify our system. The sizes of the test images range from 70*100 to 768*576 pixels. In these test images, human faces were presented in various environments. Among them, only 28 faces cannot be found correctly. Experimental results demonstrate that an approximately 97.5% (1267/1300 = 97.5%) success rate is achieved and the relative false reject rate is below 2.6 % (33/1300 = 2.53%). The false accepted rate (there are 8 non-human-face objects are detected as human faces) is below 0.7 % (8/1300 = 0.62%). The experimental sets are described as follows. Fig. 5 Experimental results with different conditions: various pose, orientation, facial hair, variation in lighting conditions, partial occlusion, presence of glasses, and diverse hair styles. Fig. 6 displays experimental results without/with skin color segmentation in complex backgrounds. 6(a) is the original input RGB color image, 6(a_Lin) is the result of binary image of 6(a) by using the scheme without skin color segmentation in Lin [7], and the number of blocks is 99; 6(a_Proposed) is the result of the skin color segmentation; 6(b) is the result of binary image of 6(a_Proposed) by using the scheme in the proposed system with skin color segmentation, and the number of blocks is only 7; 6(c) is the result of triangle; 6(d) is the potential facial region of 6(b); 6(e) is the potential facial region of 6(a_Proposed); 6(f) is the potential facial region with normalized size of 6(d); 6(g) displays the result of human face detection of 6(a_Proposed); 6(h) shows the final result of human face detection of 6(a). Due to the decrease of the number of blocks (from 99 to 7), we can expect that the speed will be improved drastically in the complicated background case. Fig. 6 shows a images with 70*161 pixels, and it needs 387.3750 seconds in Lin [7] (the number of blocks is 99) to locate the correct face position by using a P4 CPU 3.0 GHz PC. In this proposed work to locate the correct face position need only 0.0625 seconds. Moreover, the proposed scheme is significantly faster than Rowley [8] that adopted a small window (20*20) to slide over all portions of an image at various scales is a time-consuming procedure.

5. Conclusions

Since human skin color segmentation can remove complex backgrounds, it can significantly decrease execution time in the complex background case. The proposed system has two novel distributions. First, the output of human skin color segmentation must include all apparently similar skin color (e.g. as illustrated in Fig. 3) unlike in other skin color segmentation rules, so make the proposed system very robust for various illumination conditions. Second, using a color and triangle-based segmentation process can reduce the background part of a cluttered image by up to 98%. This process significantly speeds up the subsequent face detection procedure, since only 2-9% of the original image remains for further processing. The experimental results demonstrate that the proposed method is better than Rowley [6] in terms of efficiency because they adopt a small window (20 * 20) to slide over all portions of an image at various scales; the inefficient search is a time-consuming procedure resulting in loss of valuable computation time. Moreover, the proposed algorithm is significantly faster than Lin [5] in complex backgrounds as shown in Fig. 6. In the future, we plan to use this face detection system for preprocessing to solve face recognition problems.

References

[1] E. Hjelmas and B. K. Low, ″Face detection: A survey,″ Computer Vision and Image Understanding, vol. 83, Page(s): 236–274, 2001.

[2] Ming-Hsuan Yang, D.J. Kriegman, N. Ahuja, "Detecting faces in images: a survey,″ IEEE Transactions on PAMI, Volume 24, Issue 1, Jan. 2002 Page(s): 34 – 58.

[3] E. Saber and A. M. Tekalp, "Frontal-view face detection and facial feature extraction using

color, shape and symmetry based cost functions," Pattern Recognition Letters, Vol.: 19, Page(s): 669-680 (1998).

[4] Kyung-Min Cho, Jeong-Hun Jang, Ki-Sang Hong, "Adaptive skin-color filter," Pattern Recognition, Vol.: 34, Page(s): 1067-1073.

[5] C. Lin and K. C. Fan, "Triangle-based Approach to the Detection of Human Face," Pattern Recognition, Vol. 34, No. 6, Page(s): 1271-1284.

[6] H. Rowley, A. S. Baluja, and T. Kanade, ″Neural network-based face detection,″ IEEE Transactions on PAMI, Volume 20, Issue 1, Jan. 1998 Page(s):23 – 38.

[7] B. Heisele, T. Serre, S. Prentice, T. Poggio, Hierarchical classification and feature reduction for fast face detection with support vector machines, Pattern Recognition 36 (9) (2003) Page(s): 2007–2017.

[8] C. Liu, Gabor-based kernel PCA with fractional power polynomial models for face recognition, IEEE Transactions on PAMI 26 (5) (2004) Page(s):572–581.

[9] Peichung Shih and Chengjun Liu, ″ Face detection using discriminating feature analysis and Support Vector Machine,″ Pattern Recognition, Vol. 39, No. 6, (2006) Page(s): 260-276.

[10] Rafael C. Gonzalez and Richard E. Woods, "Digital Image Processing." copyright © 1992 by Addison-Wesley Publishing Company, Inc.

[11] A.M. Martinez and R. Benavente, ″The AR Face Database, ″ CVC Technical Report #24, June 1998.

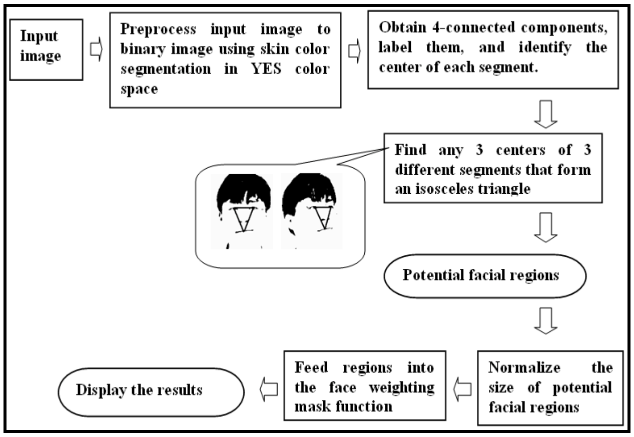
Input image
Preprocess input image to binary image using skin color segmentation in YES color space
Obtain 4-connected components, label them, and identify the center of each segment.
Find any 3 centers of 3 different segments that form an isosceles triangle
Potential facial regions
Normalize the size of potential facial regions
Feed regions into the face weighting mask function
Display the results

Fig. 1: Overview of our system

$$\begin{bmatrix} Y \\ E \\ S \end{bmatrix} = \begin{bmatrix} 0.253 & 0.684 & 0.063 \\ 0.500 & -0.500 & 0.000 \\ 0.250 & 0.250 & -0.500 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Fig. 2: RGB - YES Conversion Matrix

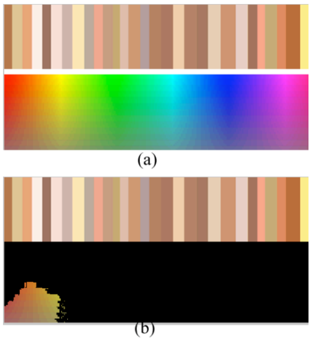

Fig. 3: Examples illustrate our skin color segmentation scheme is robust to different skin colors.

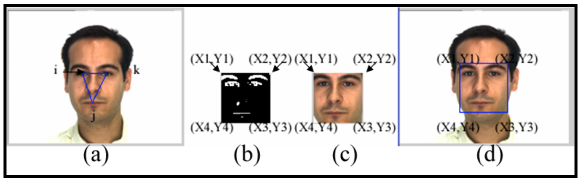

Fig. 4: The isosceles triangle i j k. (b) Three points (i, j, and k) satisfy the matching rules, which will construct an isosceles triangle.

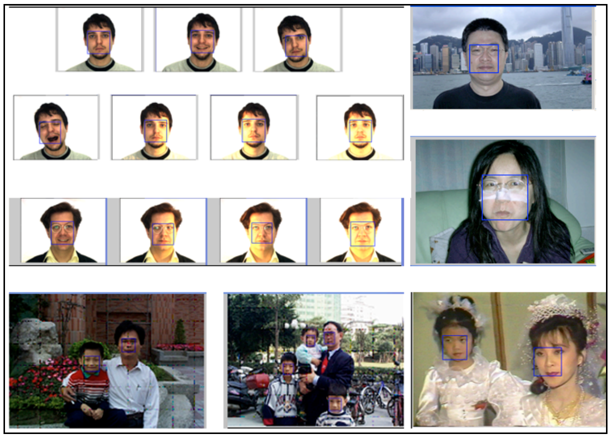

Fig. 5: Verification of face images with altered illumination conditions and different expression problems.

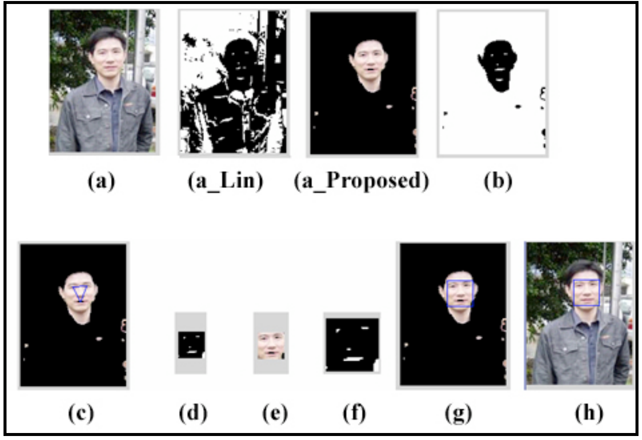

Fig. 6: Experimental results without/with skin color segmentation in complex backgrounds.