Face Detection Based on Skin Color Segmentation and AdaBoost Algorithm

Guanghui Xu, Yingcai Xiao, Shuai Xie, Sen Zhu
College of Communications Engineering, PLA University of Science and Technology
Nanjing, China
sunshine138@139.com; xiaoyc003@126.com; 1241931339@qq.com; perfect_zs@126.com

Abstract- In practical applications, face detection based on AdaBoost algorithm usually has high false positive rate and missing rate due to the interference of complex background in color images. To address these problems, this paper proposes a face detection method combined skin color segmentation with the AdaBoost algorithm. Firstly, in order to avoid the influence of poor lighting conditions, we use the Reference White algorithm to compensate the illumination of the input image. Then we convert the color space from RGB to YCgCr and segment the image into skin regions and non-skin regions. The AdaBoost algorithm is applied to train classifiers. After the skin color segmentation, the trained classifiers are used to detect faces. The experimental results show that this combined method can effectively reduce the rate of both false and miss detection.

Keywords- face detection; skin color segmentation; YCgCr color space; AdaBoost algorithm

I. INTRODUCTION

In recent years, face recognition is becoming more and more popular as a means of identity authentication applications, and has become a hotspot of current academic research. Face detection is the prophase of face recognition, so it also attracts more and more attention.

In 2001, Paul Viola and Michael Jones proposed a method based on AdaBoost algorithm for face detection which can detect face in real time [1]. They compute Haar-like features by using integral image and detect faces by training cascade classifier. This method made face detection can be applied to real-life situations and has had a profound impact on the development of face detection technology. However, when the background of the image is complex, this algorithm has a high false positive rate or missing rate [2].

For human face detection, skin color is a significant feature. The YCgCr color space has good clustering performance and we can easily detect skin-like regions by thresholding method in this color space. So we can first use the skin color segmentation based method to exclude the interference of background regions.

In this paper, we use a method which combined skin color segmentation with the AdaBoost algorithm to detect faces. The input image is first segmented into skin-like regions and non-skin regions in the YCgCr color space, then use the trained classifiers to detect faces in skin regions. With the skin color segmentation, we can distinguish skin-like regions and background regions. Therefore, the detection accuracy and speed could be improved.

II. COLOR SPACE MODEL

In real life, face detection is mostly used in real-time scenes, so its algorithms must be has low computing complexity and can run as fast as possible [3, 4]. Skin color is a simple but effective feature of human face. The skin color based algorithms usually processing faster than that based on many other facial features, so we can use some skin color based methods to detect human faces.

The key of skin color segmentation is to choose a suitable skin color space. There are many different color space models used for skin color segmentation, such as HSI, RGB, CIELab, YIQ, YCbCr, YCgCr, etc. [5, 6]. Here we introduce RGB, YCbCr and YCgCr these three color spaces.

A. RGB Color Space Model

The RGB color space is a most widely used color space for digital image storing and processing. The chrominance and luminance component are mixed in RGB color space, it is difficult to separate them [7]. Moreover, RGB color space is device-dependent, the same image may look different on different devices [8]. For these reasons, RGB color space is not appropriate for skin color segmentation algorithm.

B. YCbCr Color Space Model

The YCbCr color space has good performance of clustering, and the transformation from RGB to YCbCr is linear. More importantly, different from RGB color space, the chrominance component and luminance component in YCbCr color space are separated explicitly [9], so it is appropriate for skin color segmentation. Y indicates the luminance component and it is the weighted sum of RGB values. Cr and Cb are the red-difference and the blue-difference chrominance components [10]. The conversion formula of transformation from RGB to YCbCr is as follows:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 0.256 & 0.5041 & 0.0979 \\ -0.1482 & -0.291 & 0.4392 \\ 0.4392 & -0.3677 & -0.0714 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

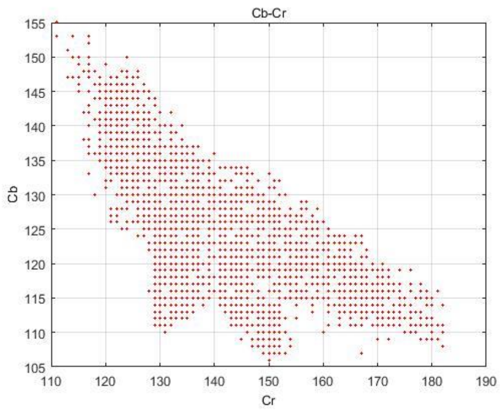
C. YCgCr Color Space Model

In YCbCr space, the Cb component is the difference between the luminance component and the blue component.

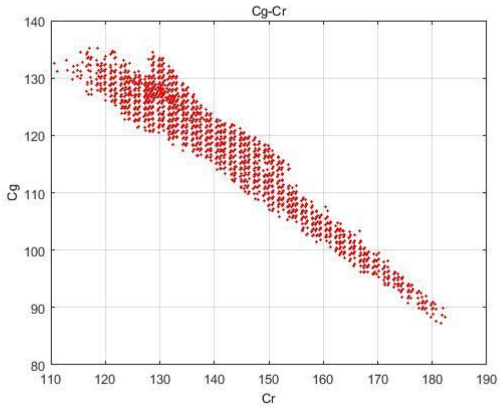
However, the blue component occupies a smaller proportion in the skin color. Therefore, De Dios proposed the YCgCr color space that Cg indicates the difference between the green component and the luminance component. This is more significant for skin color segmentation [11]. The conversion formula of transformation from RGB to YCgCr is as follows:

$$\begin{bmatrix} Y \\ Cg \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 0.256 & 0.5041 & 0.0979 \\ -0.318 & -0.4392 & -0.1212 \\ 0.4392 & -0.3677 & -0.0714 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2)$$

Fig. 1 is the distribution of an image in Cb-Cr plane and Cg-Cr plane. It is obviously that YCgCr color space has better clustering performance than YCbCr color space.



(a)



(b)

Fig.1. (a) Distribution in Cb-Cr plane. (b) Distribution in Cg-Cr plane.

III. Skin Color Segmentation

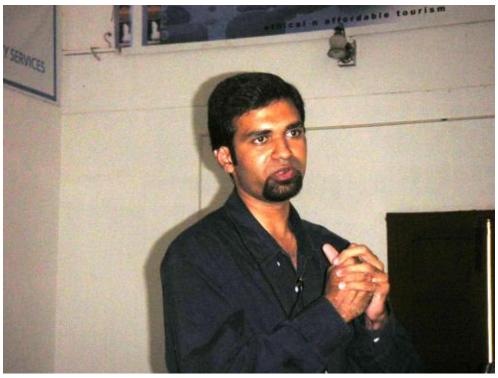
A. Lighting Compensation

The effect of skin color segmentation is often affected by the lighting conditions. In the case of poor lighting conditions, we will not get the desired results. In order to eliminate this adverse influence, we need to compensate the illumination of the image before using skin color segmentation algorithms. There are three commonly used methods of lighting compensation: histogram equalization, Gray World Theory [12] and Reference White [13]. We adopt the Reference White method because this method can achieve better results than the other two methods. The details of the Reference White algorithm are described as the following steps:

- Traverse all the pixels in the image and arrange them according to the brightness from top to low.
- Take pixels with top 5% of the luminance values as the reference white and set these pixels to white (R=G=B=255).
- Calculate the compensation coefficient $k = 255/v$, where v is the mean value of the original value of the reference-white pixels.
- Multiply the compensation coefficient by the original value to adjust the value of the other pixels.

Through the above steps, the image with poor lighting conditions will be compensated. The result of lighting compensation by Reference White algorithm is shown in Fig. 2. As we can see, the original image with poor lighting conditions can be compensated well.

(a)



(b)

Fig.2. (a) The original image. (b) The result image after lighting compensation.

B. Skin Detection

After the process of lighting compensation, the color space is converted from RGB to YCgCr so that we can segment the image into skin-like regions and non-skin regions in this space. Thresholding segmentation is the simplest method, which has better computing performance than the Gaussian skin color model. We regard a pixel as skin if it satisfies the following conditions:

$$\begin{cases} Y > 80 \\ 100 < Cg < 130 \\ 135 < Cr < 175 \end{cases} \quad (3)$$

The result of thresholding segmentation is a binary image where white represent the skin-like regions and black represent the non-skin regions. Then we use the median filter to remove noise. The median filter can remove noise with less attenuation of edges [14]. Fig. 3 is the results of skin color segmentation.

(a) (b)

(c) (d)

Fig.3. (a) The original image. (b) The lighting compensated image. (c) The segmentation result of (a). (d) The segmentation result of (b).

C. Morphological Operation

We get a binary image after threshold segmentation. For better performance, it is necessary to use morphological operations. The morphological close and open operation is applied. Moreover, holes filling operation is also needed because the facial regions may contain holes which represent eyes, mouth, facial hair, etc. The results of morphological operations are showed as Fig. 4.

(a) (b)

(c) (d)

Fig.4. (a) Fig. 3(c) after morphological operations. (b) Fig. 3(d) after morphological operations. (c) The final image of Fig. 3(a). (d) The final image of Fig. 3(b).

IV. Face Detection using AdaBoost Algorithm

In this paper, we first obtain face region candidates by the skin color segmentation method. After that, we use the AdaBoost algorithm to distinguish human faces and non-human faces.

AdaBoost algorithm was firstly proposed by Freund and Schapire in 1995. In 2001, Viola and Jones applied this algorithm to detect faces and successfully constructed a detection system which can detect human face in real time [15]. They use the integral image to solve the speed problem and uses cascade classifier to achieve high detection rates. The details of this algorithm are described as the following subsections.

A. Harr-Like Features

Haar-like feature are rectangular digital image features. A rectangular Haar-like feature is defined as the difference of the sum of pixels in white rectangles and the sum of pixels in black rectangles [16]. The typical rectangular features are as shown in Fig. 5:

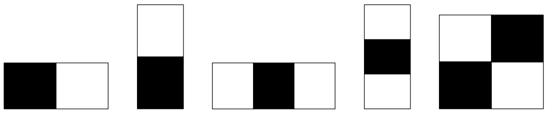


Fig.5. Haar-like feature.

Viola and Jones use integral image to calculate the value of rectangular features. The value of coordinate (x, y) in an integral image is the sum of all pixels which located on the up and left region of the original image at coordinate (x, y).

$$I_i(x, y) = \sum_{x' \le x, y' \le y} I_o(x', y') \quad (4)$$

where $ii(x, y)$ and $I(x, y)$ respectively are the pixels value of integral image at coordinate (x, y) and the pixel value of original image at coordinate (x, y). Due to the use of the integral image, the value of Haar-like feature can be calculated very fast only by using addition and subtraction operation [17].

B. AdaBoost Algorithm

There are a lot of Haar-like features in an image and we use the AdaBoost algorithm to select the best of them as weak classifiers. The details of the AdaBoost algorithm are as follows [18]:

1) Giving $(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)$, where $x_i \in X$ are the training image samples and $y_i \in Y = \{-1, 1\}$. $y_i = -1$ represents negative sample and $y_i = 1$ represents positive sample.

2) For n training samples, initializes weights $w_{1,t} = 1/n$.

3) For $t = 1, \cdots, T$
   - Normalize the weights:

$$w_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^{n} w_{t,j}}$$

   - For each feature j, train a weak classifier $h_j(x)$ in order to make the error function ε_j obtains a minimum value.

$$\varepsilon_j = \sum_{i=1}^{n} w_{t,i} \left| h_j(x_i) - y_i \right|$$

   - Choose the weak classifier h_t which with the lowest error ε_t.
   - Update the weights $w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$, where $e_i = 0$ if x_i is classified correctly, otherwise $e_i = 1$, and $\beta_t = \frac{\varepsilon_t}{1-\varepsilon_t}$.
   - The final strong classifier is:

$$h(x) = \begin{cases} 1, & \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0, & otherwise \end{cases} \quad \text{where } \alpha_t = \log \frac{1}{\beta_t}.$$

For each round, this algorithm selects a best feature as a weak classifier. The final strong classifier is the linear combination of these weak classifiers.

C. Cascaded Classifier

As shown in Fig. 6, cascaded classifier is composed of a series of strong classifier. The strong classifiers at front usually contain less weak classifiers, so it can exclude the most part of non-face areas rapidly. The strong classifiers at rearward position contain more weak classifiers than the front strong classifiers, so that they can distinguish the difficult samples.

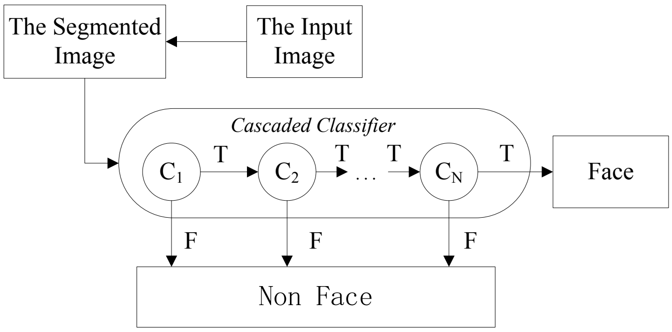


Fig.6. Cascade Classifier.

After skin color segmentation, the image is sent to the cascaded classifier to detect faces. If the first strong classifier judges the image as no face, then terminate the detection process. Otherwise the image will be sent to the next strong classifier for further verifying.

V. EXPERIMENTAL RESULTS AND DISCUSSION

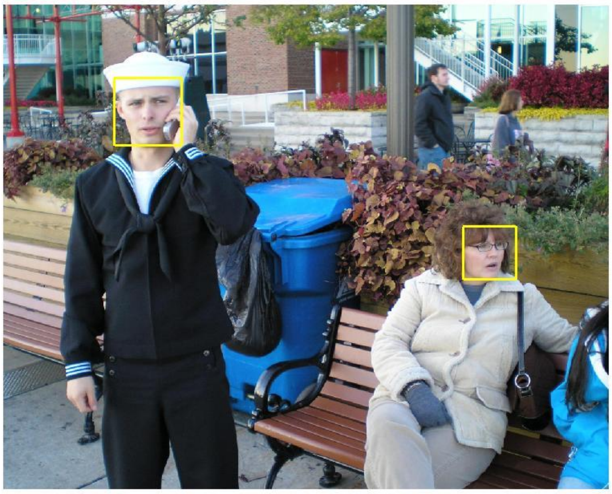
In this paper, we implement the proposed face detection algorithm in MATLAB R2014a. We use the MIT face database to train classifier, and use the AFW database and Wider Face database to test our algorithm. Fig. 7 and Fig. 8 show partial experimental results.

(a)

(b)

Fig.7. (a) Face detection by AdaBoost algorithm. (b) Face detection by skin color segmentation and AdaBoost algorithm.

(a)



(b)

Fig.8. (a) Face detection by AdaBoost algorithm. (b) Face detection by skin color segmentation and AdaBoost algorithm.

From Fig. 7 and Fig. 8 we can see that the combination of skin color segmentation and AdaBoost algorithm can effectively improve the accuracy of face detection. Table I is the statistical results of our experiment.

TABLE I. STATISTICAL RESULTS OF OUR EXPERIMENT

Method	AdaBoost	Combined Method
Face number	356	356
Hit	319	328
False positive	15	6
Hit rate	89.61%	92.13%
False positive rate	4.2%	1.8%

VI. CONCLUSION

In this paper, a face detection method which combines skin color segmentation and AdaBoost algorithm together is described in detail. Skin color segmentation can detect the skin-like regions and reject other background regions. We use the AFW database and Wider Face database which images both have complex background to test this method. The experimental results show that the proposed method can effectively reduce the rate of both false and miss detection.

REFERENCES

[1] P. Viola and M. Jones. “Rapid object detection using a boosted cascade of simple features,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001. Vol. 1. IEEE, 2001.

[2] Li Z, Xue L, Tan F. “Face detection in complex background based on skin color features and improved AdaBoost algorithms,” 2010 IEEE International Conference on Progress in Informatics and Computing. IEEE, 2010, 2: 723-727.

[3] Kim J M, Kim K H, Song M K. “Real Time Face Detection and Recognition Using Rectangular Feature Based Classifier and Modified Matching Algorithm,” 2009 Fifth International Conference on Natural Computation. IEEE, 2009, 2: 171-175.

[4] Dhivakar B, Sridevi C, Selvakumar S, et al. “Face detection and recognition using skin color,” 2015 3rd International Conference on Signal Processing, Communication and Networking. IEEE, 2015: 1-7.

[5] Chaves-González J M, Vega-Rodríguez M A, Gómez-Pulido J A, et al. “Detecting skin in face recognition systems: A colour spaces study,” Digital Signal Processing, 2010, 20(3): 806-823.

[6] Daithankar M V, Karande K J, Harale A D. “Analysis of skin color models for face detection,” 2014 International Conference on Communications and Signal Processing. IEEE, 2014: 533-537.

[7] Mohanty R, Raghunadh M V. “Skin Color Segmentation based Face Detection using Multi-Color Space,” International Journal of Advanced Research in Computer and Communication Engineering, 2016, 5(5):470-475.

[8] Kaur A, Kranthi B V. “Comparison between YCbCr color space and CIELab color space for skin color segmentation,” International Journal of Applied Information Systems, 2012, 3(4): 30-33.

[9] Archana T, Gopal T V, Kumar M P. “Skin Color Based Segmentation for Multiple Face Detection,” Journal of Innovation in Computer Science and Engineering, 2015, 4(2):41-47.

[10] Kumar P, Shashidhara M. “Skin color segmentation for detecting human face region in image,” 2014 International Conference on Communications and Signal Processing. IEEE, 2014: 001-005.

[11] De Dios J J, García N. “Face detection based on a new color space YCgCr,” IEEE International Conference on Image Processing. IEEE, 2003, 3: III-909-12 vol. 2.

[12] Huang H M, Liu H S, Liu G P. “Real-time Face Tracking in Video Using Gray World Assumption for Images Preprocess,” International Journal of Digital Content Technology and its Applications, 2013, 7(12): 175.

[13] Hsu R L, Abdel-Mottaleb M, Jain A K. “Face detection in color images,” International Conference on Image Processing, 2001. Proceedings IEEE, 2001, 1: 1046-1049.

[14] Alabbasi H A, Moldoveanu F. “Human face detection from images based on skin color,” 2014 18th International Conference on System Theory, Control and Computing. IEEE, 2014: 532-537.

[15] Krishna M G, Srinivasulu A. “Face detection system on AdaBoost algorithm using Haar classifiers,” International Journal of Modern Engineering Research, 2012, 2(5): 3556-3560.

[16] Wang K, Zhang H. “Research on an improved algorithm of face detection based on skin color features and cascaded AdaBoost,” 2015 IEEE 16th International Conference on Communication Technology. IEEE, 2015: 779-782.

[17] Tofighi A, Monadjemi S A. “Face detection and recognition using skin color and AdaBoost algorithm combined with Gabor features and SVM classifier,” 2011 International Conference on Multimedia and Signal Processing. IEEE, 2011, 1: 141-145.

[18] Viola P, Jones M J. “Robust real-time face detection,” International journal of computer vision, 2004, 57(2): 137-154.