A New Method for Eye Detection in Color Images

Mohammadreza Ramezanpour
Department of Computer Science & Research Branch Azad University, Arak.Iran
E-mail: Mr.ramezanpoor@gmail.com

Mohamad Ali Azimi
Department of Computer Science & Research Branch Azad University Dezful,.Iran
E-mail: Azimi.kashani@gmail.com

Mohammad Rahmati
Computer Engineering Amirkabir University of Technology, Tehran, Iran
E-mail: rahmati@aut.ac.ir

Abstract

The problem of eye detection in face images is very important for a large number of applications ranging from face recognition to gaze tracking. In this paper we propose a new algorithm for eyes detection. First, the face region is extracted from the image by skin-color information. Second, horizontal projection in image is used to approximate region of the eye be obtained . At last, the eye center is located by a corner detection algorithm. Part of the FACE94 Database has been used for experiments and benchmarking. Experimental results show that a correct eye detection rate of 94% can be achieved on 300 FACE94 images with different facial expressions and lighting conditions.

Keywords: *Eye detection, face region, Eigen value.*

1. Introduction

Automatic tracking of eyes and gaze direction is an interesting topic in computer vision with its application in biometric, security, intelligent human-computer interfaces, and driver's drowsiness detection system[1]. Localization and extraction of eyes are operations requisite for solving problem. In [2] eye localization methods have been classified into five main categories: 1) template matching methods in which first construct an eye template, and then compare it with sub-images in the areas to be detected. The sub-image with the highest matching score is considered to be the eye centre. 2) characteristic detection methods which use edges, corners, intensity or other characteristics of the face to locate eyes. 3) machine learning methods which set optimal classify plane to distinguish eye and non-eye regions. 4) active IR based methods in which use near infrared camera to generate images with bright pupil to help detecting eyes. 5) image processing methods which locate eyes by dealing with intensity and color information. In [3], skin-color filter is used to extract face. The eye position is gained by gradient characteristic projection and corresponding conditions setting. In [4] use the Haar-like features to detect the eye. This method trains classifiers with a few thousands of sample view images of object and construct a cascade of classifiers to detect eye rapidly.

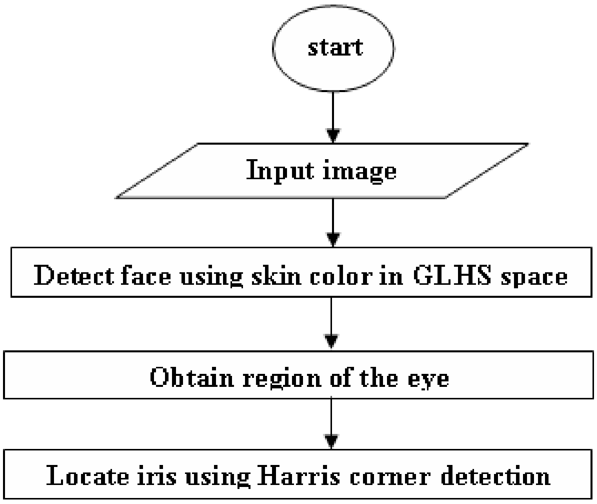

Figure 1. Algorithm of eye detection

2. Skin Color Segmentation

Skin color model is the mathematical model that describes the distribution of skin colors. A proper color space should be selected before constructing skin color models. There are many color spaces. The RGB expression we often use is not proper for skin model, because in RGB space, three-based color (r, g ,b) presents not only color but also luminance. In this paper we choose a improved GLHS space [6] as mapping space because this color model is robust against different types of skin. GLSH' color space has a non-linear relationship with the RGB space, so it is a complex process from RGB color space to GLHS' color space.

If

$$\begin{aligned} \min(c) &= \min(R,G,B)/255, \\ mid(c) &= mid(R,G,B)/255, \\ \max(c) &= \max(R,G,B)/255 \end{aligned} \tag{1}$$

and the lightness, hue and saturation can be calculated as follows:

Hue:

$$h(c) = k(c) + f(c) \tag{2}$$

where k(c) denotes the sequence number of the subspace c belongs to and f(c) is used to calculate the angle of c in the subspace.

$$k(c) = \begin{cases} 0: R > G \geq B \\ 1: G \geq R > B \\ 2: G > B \geq R \\ 3: B \geq G > R \\ 4: B > R \geq G \\ 5: R \geq B > G \end{cases} \tag{3}$$

$$f(c)=\begin{cases}\dfrac{mid(c)-\min(c)}{\max(c)-\min(c)} : k(c) \text{ is even} \\ \dfrac{\max(c)-mid(c)}{\max(c)-\min(c)} : k(c) \text{ is odd}\end{cases} \tag{4}$$

Lightness:

$$l(c)=(\max(c)-mid(c))-(mid(c)-\min(c)) \tag{5}$$

Saturation:

$$s(c)=\frac{1}{3}(\max(c)-\min(c))+\frac{1}{3}(mid(c)-\min(c)) \tag{6}$$

Good clustering performance can be achieved when the following criteria are satisfied.

$$\begin{cases}0.065 \le S(c) \le 0.25 \\ -0.15 \le l(c) \le 0.27 \\ 2 \le h(c) \le 6 \\ R > 90\end{cases} \tag{7}$$

3. Corner Detection

The Corner Detection finds corners in an image using the Harris corner detection, minimum eigenvalue, or local intensity comparison method. The Corner Detection finds the corners in the image based on the pixels that have the largest corner metric values.

For the most accurate results, use the minimum eigenvalue method. For the fastest computation, use the Local Intensity Comparison. For the trade-off between accuracy and computation, use the Harris Corner Detection Method.

3.1. Minimum Eigenvalue Method

This method is more computationally expensive than the Harris corner detection algorithm because it directly calculates the eigenvalues of the sum of the squared difference matrix, M. The sum of the squared difference matrix, M, is defined as follows:

$$M=\begin{bmatrix} A & C \\ C & B \end{bmatrix}$$

The previous equation is based on the following values:

$$A=(I_x)^2 \otimes w$$
$$B=(I_y)^2 \otimes w$$
$$C=(I_x I_y)^2 \otimes w$$

where I_x and I_y are the gradients of the input image, I, in the x and y direction, respectively. The $\otimes$ symbol denotes a convolution operation. Use the Coefficients for separable smoothing filter parameter to define a vector of filter coefficients. This vector of coefficients is multiplied by its transpose to create a matrix of filter coefficients, w. Smaller eigenvalue calculates the sum of the squared difference matrix. This minimum eigenvalue corresponds to the corner metric matrix.

3.2 Harris Corner Detection Method

The Harris corner detection method avoids the explicit computation of the eigenvalues of the sum of squared differences matrix by solving for the following corner metric matrix, R:

$$R = AB - C^2 - K(A+B)^2 \tag{8}$$

A, B, C are defined in the previous section, Minimum Eigen value Method. The variable K corresponds to the sensitivity factor. Its value can be specified by using the Sensitivity factor (0<K<0.25) parameter. The smaller the value of k, the more likely it is that the algorithm can detect sharp corners. Use the Coefficients for separable smoothing filter parameter to define a vector of filter coefficients. This vector of coefficients is multiplied by its transpose to create a matrix of filter coefficients, w.

3.3 Local Intensity Comparison

This method determines that a pixel is a possible corner if it has either, N contiguous valid bright surrounding pixels, or N contiguous dark surrounding pixels. Specifying the value of N is discussed later in this section. The next section explains how these surrounding pixels are found. We suppose that p is the pixel under consideration and j is one of the pixels surrounding p. The locations of the other surrounding pixels are denoted by the shaded areas in fig 2.

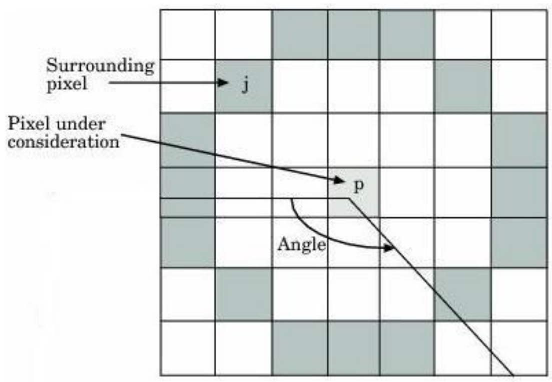

Figure 2. Locations of the other surrounding pixels are denoted by the shaded areas

I_p and I_j are the intensities of pixels *p* and *j*, respectively. Pixel *j* is a valid bright surrounding pixel if $I_j - I_p \geq T$. Similarly, pixel *j* is a valid dark surrounding pixel if $I_p - I_j \geq T$. In these equations, T is the value you specified for the Intensity comparison threshold parameter. This process repeats to determine whether it has N contiguous valid surrounding pixels. The value of N is related to the value you specify for the Maximum angle to be considered a corner (in degrees), as shown in Table 1.

Table 1. The value of N is related to the value you specify for the maximum angle .

Number of Valid Surrounding Pixels, N	Angle (degrees)
15	22.5
14	45
13	67.5
12	90
11	112.5
10	135
9	157.5

After that a pixel determines is a possible corner, it computes its corner metric using the following equation:

$$R = \max\begin{pmatrix} \sum_{j:I_j \geq I_p - T} \left| I_p - I_j \right| - T, \\ \sum_{j:I_j \leq I_p - T} \left| I_p - I_j \right| - T \end{pmatrix} \tag{9}$$

Figure 3. Original image

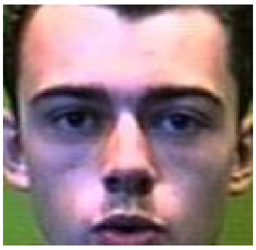

Figure 4. Face detection

4. Eye Region Detection

As explained in section 2,the face region is extracted from the original image (fig 3) by skin-color information in GLHS color space (fig 4) .To obtain the exact location of the eye, therefore approximate region of the eye should be obtained by horizontal projection . The sum of each row is then horizontally plotted (fig5). Due to the fact that the iris has lower intensity than the eye white, it can be claimed that the minimum point on the diagram is related to iris region. But the above hypothesis will not work correctly

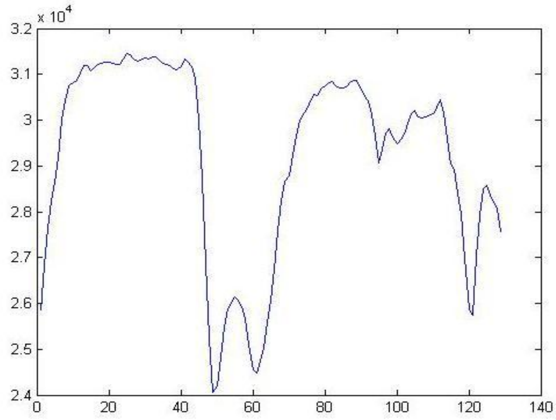

Figure 5. Horizonal projection of face region

This rule that the top eye region and low region around the eyes are brighter than the eye itself and local maximum points on this diagram are related to these two region . Now it can be claimed that the eye is between these two maximum points in diagram. Based on the above mentioned, the eye region could be cropped from the face region (Fig 6).Now we use Harris corner detection to detect corners in output image of the previous stage. Due to the fact that the iris is darker than the eye white , it detects irises as corners(fig 7). Thus the place of iris is diagnosed in the image.

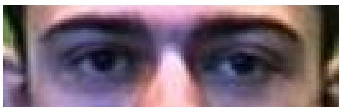

Figure 6. Eye region cropped from the face region

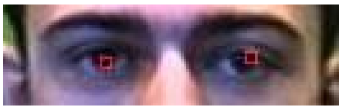

Figure 7. Iris detect with Harris corner detection

5. Experimental Result

Color images downloaded from[6] are chosen to evaluate this algorithm. Specify the sensitivity factor, k. The smaller the value of k the more likely the algorithm is to detect sharp corners. K has selected 0.01 for this experiment.

In fig 8 and fig 9 some results of correct detection are shown, in particular fig 8 shown the eye detection on image of people wearing glasses.

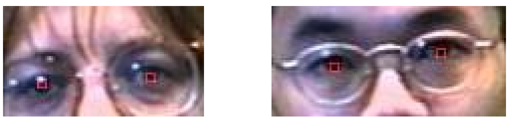

Figure 8. Some result with people wearing glasses

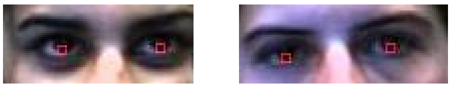

Figure 9. Some result

6. Conclusions

In this paper a new algorithm for eye detection is proposed .First, face is detected in image. Then approximate region of the eye should be obtained by horizontal projection. The eye center is located by Harris corner detection algorithm . Due to using of GLHS color space, lighting condition don't affect on algorithm output. Different experiments, carried out on images of subjects with different facial expressions, some of them wearing glasses, demonstrate the effectiveness and robustness of the proposed algorithm. The eyes could be correctly located in 94% of the images.

References

[1] Xiaoyun Deng, Chip-Hong Chang ,Erwin "A New Method for Eye Extraction from Facial Image ", *IEEE International Workshop on Electronic Design*, Test and Applications 2004

[2] Wen-Tao Wang, Chao Xu, Hong-Wei Shen, "EYE LOCALIZATION BASED ON HUE IMAGE PROCESSING", *IEEE International Symposium on Intelligent Signal Processing and Communication Systems*, Nov.28-Dec.1, 2007.

[3] Xiaoyun Deng, Chip-Hong Chang, Erwin Brandle, "A new method for eye extraction from facial image", *IEEE International Workshop on Electronic Design*, Test and Applications, pp. 29-34,Jan 2004.

[4] Ba Linh NGUYEN, "Eye Gaze Tracking", *IEEE*, vol. 2, 2009.

[5] Wei-Min Zheng, Zhe-Ming Lu, Xiu-Na Xu, "A Novel Skin Clustering Method for Face Detection", *First International Conference on Innovative Computing, Information and Control*, 2006 Page(s):166 - 169 Vol.

[6] http://cswww.essex.ac.uk/mv/allfaces/faces94.html.