

Iris Segmentation Based on Ellipse Detection for Gaze Tracking System

Yumeng Zhang^{1, a}, Cai Meng^{2, b}

¹School of Beihang University, Beijing 100191, China;

²School of Beihang University, Beijing 100191, China.

^azhymeng93@gmail.com, ^bTsai@buaa.edu.cn

Abstract: This paper propose a novel algorithm to detect the center and boundary of iris in face images of low resolution for gaze tracking system. The algorithm localizes the eye accurately by face alignments and uses convolution to detect the approximate pixel of the iris center which can determines the iris region. After thresholding and a series of morphological processing, the boundary and precise center of iris is detected via ellipse detection. The experiment shows good result on different kinds of people looking into different directions on Columbia Gaze Data Set.

Keywords: Face Alignment, Iris Detection, Morphological Processing, Ellipse Detection.

1. Introduction

The eye is considered to be the most salient features of human beings, because it not only accepts 80% of the information a person gets every day, but expresses a person's emotional state, needs and desire[1]. Therefore, gaze tracking system attracts more and more attention in the last decades. As the important prerequisite of gaze tracking, much effort has been devoted to improving the performance of iris detection too.

In recent years, iris detection research into low-cost images based on web-cameras has become a promising endeavor for scientists. The taxonomy of iris detection technics consists of model-based, feature-based and hybrid methods. The model-based methods utilize the whole appearance of the eye [2] [3] [4]. These approaches train and classify a series of selected features or match learned templates from numbers of people to detect the location of the eyes of the current observer. The model-based methods are robust but not sensible to subtle eye center movements. In contrast, feature-based methods utilize visual features such as corners, edges and gradients of the current observer to detect the iris [5] [6] [7]. These methods may be more affected by illumination but can achieve a higher accuracy.

Among these works, most focus on detecting the iris center and has reach a high accuracy even on low-resolution image, some detect the iris boundary using a circle model which perform better on high resolution image. However, in gaze tracking system, both the center and boundary are important features of the iris. This paper proposed a feature-based method about iris detection including iris boundary and iris center in natural light and low-cost images. The remainder of this paper is organized as follows: section 2 describes our novel algorithm. The experiment results and conclusion are presented in Section 3 and section 4.

2. Proposed Method

In the proposed work, we present a novel method for detecting the boundary and center of iris. As shown in Fig.1, firstly, eye location is determined by a few landmarks based on face alignment. And then we obtain the region of iris by approximate center of center which determined via a kernel convolution. Thirdly, we segment the image of iris region by using thresholding and morphological processing. Finally, accurate contour and precise center of iris are obtained utilizing ellipse detection.

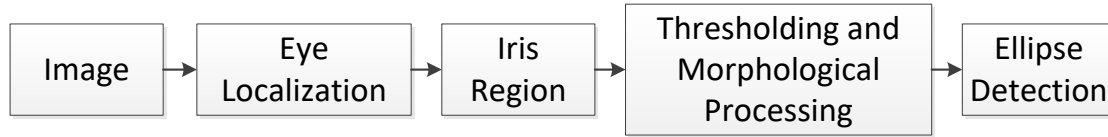


Fig. 1 Flow diagram of the method proposed.

2.1 Eye Localization

In this section, we mainly determine the exact location of the eye. The method [10] accomplishes face detection and face alignment at the same time with a high accuracy in real time. In this paper, we use the method [10] detects 68 facial landmarks including face contour, eye, nose and lips. As shown in Fig. 2, the points labeling 38-42 represent the left eye, and the points labeling 43-49 represent the right eye. Take the right eye as an example, we connect the points labeling 43-49 in turn to form a polygon. Then the exact location of the eye can be determined by finding the minimum bounding rectangle of the polygon. In this way, we can reduce the interference of eyebrows and eyelids. Fig. 3 (a) shows the result of right eye.

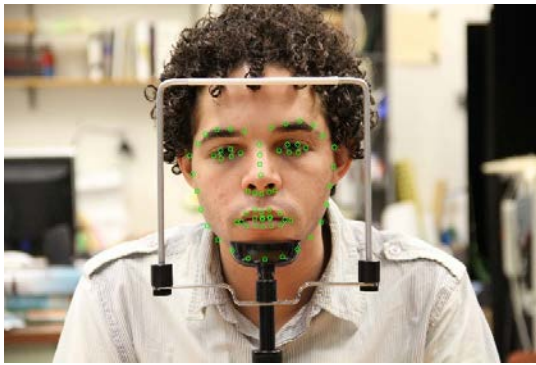


Figure. 2 Face landmark.

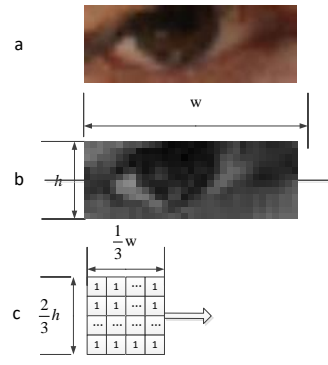


Figure. 3 Iris center estimation.

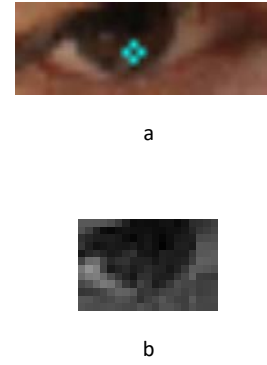


Figure. 4 Iris center and region.

(a) Eye region.

(b) Gray image of the eye.

(c) Kernel

(a) Pixel c.

(b) Iris region

2.2 Iris Region

In general, the iris is supposed to be the darkest region of the image. Since the diameter of iris is about width of the eye, we use a kernel with all the elements set to 1, where w and h are the width and height of eye region respectively. As shown in Fig. 3, (b) is the gray image of eye and (c) is the image of kernel. For each pixel on the middle line in gray image of eye, we calculate the convolution of image and kernel which all elements are 1. Then the pixel with the largest value is concerned to be the center of iris region. Hence, the center of iris region is defined,

$$c = \arg \max_{x_i} \sum_{x_i \in \varepsilon|c} I(x_i) \quad (1)$$

where x_i is pixel point on the line ε . ε is the neighborhood of c determined by the kernel. $I(x_i)$ is a pixel of the neighborhood, and $I(x_i)$ are the gray values. The center of iris region has the maximum sum of the gray values of all pixels in the neighborhood. Then the iris region is decided by the rectangle which takes c as center, h as height, w as weight, as shown in Fig. 4.

2.3 Thresholding and Morphological Processing

Because the contour detected in iris region is complex, we cannot obtain accurate boundary via 'bad' ellipse, therefore, we need to get a 'good' ellipse by thresholding and morphological processing.

Firstly we choose a threshold value to provide a satisfactory binaryzation. As shown in Fig. 4(b), iris region is composed of iris and scalar, and the iris is much darker than the scalar. So we can indicate that the gray values of the image mainly lie on two ranges, as shown in the gray histogram of eye region denoted in Fig. 5(b). Due to this feature of iris region, we select the threshold value in the following procedure. Suppose Y is a gray value between 0-255, $HistGram(Y)$ is the number of pixels whose gray values are Y . While $HistGram(Y-1) < HistGram(Y)$ and

$HistGram(Y+1) < HistGram(Y)$, we define Y as a peak value. There will be several peak values because of noise, so we smooth the histogram as (2).

$$HistGram(Y) = \begin{cases} HistGram(0) + HistGram(0) + HistGram(1) & Y = 0 \\ HistGram(Y-1) + HistGram(Y) + HistGram(Y+1) & 0 < Y < 255 \\ HistGram(254) + HistGram(255) + HistGram(255) & Y = 255 \end{cases} \quad (2)$$

Repeat this process until there are only two peak values Y_1 and Y_2 ($Y_1 < Y_2$) in the histogram. Y_1 is the value of iris, and Y_2 is the value of scalar. The histogram after smooth is shown in Fig.5 (b). Then the thresholding value T is defined,

$$T = \frac{1}{2}(Y_1 + Y_2) \quad (3)$$

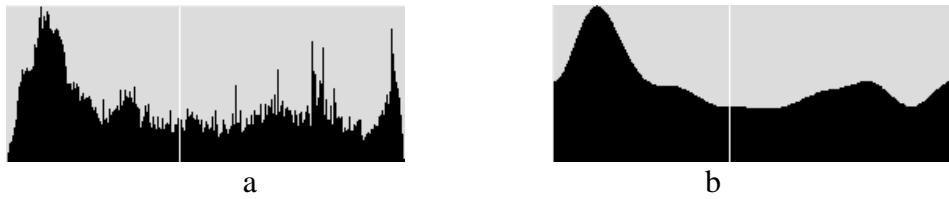


Figure 5 (a) Histogram of iris region. (b) Histogram after smooth.

As shown in Fig. 6, narrow discontinuities and fine protrusions caused by scalar and eyelids under different illumination still exist in the binary image. The situation will lead to considerable errors in the following ellipse detection. In this paper, morphological processing is used to reduce these effects. As shown in Fig. 6, we use a close operation with a 2×3 rectangle element to narrow the gap and eliminates small holes and then an open operation with same rectangle element to disconnect the narrow discontinuities and eliminate the fine protrusions. Then the iris is segmented via finding the maximum connected domain. Finally, we find the convex hull of the domain to smooth the contour.



Figure. 6 (a) Histogram. (b) Thresholding. (c) Close. (d) Open. (e) Connected domain. (f) Convex hull.

2.4 Ellipse Detection

Some researches detected the iris by a circle template, but the boundary of iris is more likely an ellipse. Because of physiological factors and the truth that the image is the projection of the iris through the camera, the iris contour is an ellipse after projection even if it is a circle. So, the contour and center of iris will be more precise by using ellipse detection instead of circle detection.

In this paper, the iris is detected in the following steps. (a) Gaussian Blur; (b) Canny detection; (c) 8 neighborhood method for edge line extraction; (d) Douglas Peucker (DP) algorithm for edge approximation; (e) Extraction of arc segment with elliptic characteristic by dividing those arc segment obtained in step (d); (f) Grouping and screening of the elliptical arc segment in step e. Fig.7 shows the result of both center and boundary of eyes in the image.

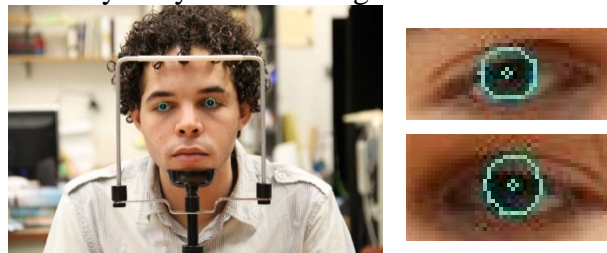


Figure. 7 Iris center and boundary.

3. Experiment

Columbia Gaze Data Set[11] contains 5880 images of 56 people over varying gaze directions and head poses. There are 5 head poses and 21 gaze directions per head pose for each subject. We select the images which are recorded by a camera in front of the head looking in all 21 gaze directions from 35 people who didn't wear glasses. Thus, there are 735 images in our experiment. We normalize all the image with a width of 640, which is the normal size of a web camera.

Fig. 8 shows the satisfactory results of our method working on both males and females from different race of people, and Fig. 9 shows the results of a subject from different directions.

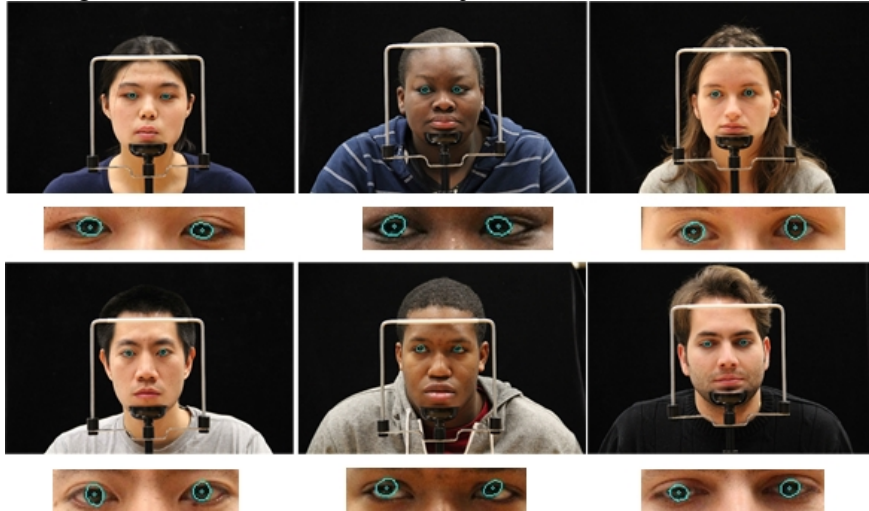


Figure. 8 Results on different kinds of people.

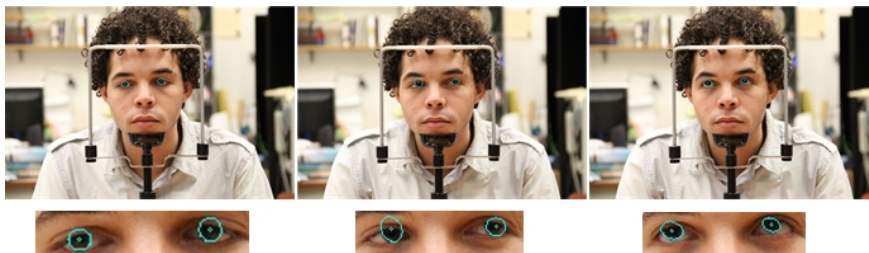


Figure. 9 Results on different directions.

However, the method fails in some cases. In the first case, we can get the pixel correctly, but cannot detect the ellipse of the iris boundary due to the not-wide-open eye, so we take as the center of the iris, as shown in Fig. 10 (a) (left eye). In the second case, we cannot even obtain the right because the gray difference is not obvious in the eye caused by illumination.

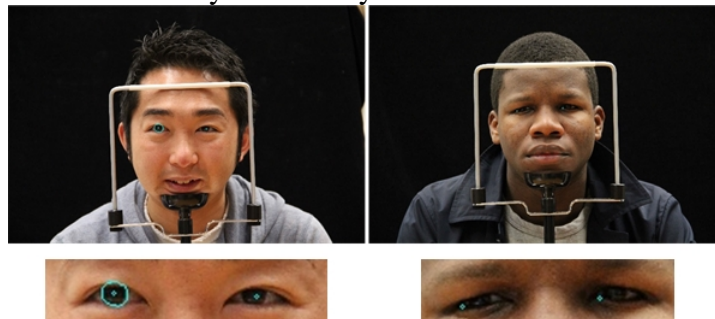


Figure. 10 (a) Right center and no boundary. (b) Wrong center.

4. Conclusions

This paper represents an algorithm to detect both the center and boundary of iris in low resolution images for gaze tracking system. We tested the method on public available Columbia Gaze Data Set.

The results are satisfactory on different people and different directions, however, the robustness to illumination can be improved in the future.

References

- [1] Jesorsky O, Kirchberg K J, Frischholz R. Robust Face Detection Using the Hausdorff Distance, C, International Conference on Audio- and Video-Based Biometric Person Authentication. Springer-Verlag, (2001) 90-95
- [2] Kroon B, Hanjalic A, Maas S M P. Eye localization for face matching: is it always useful and under what conditions?, C, ACM International Conference on Image and Video Retrieval, Civr 2008, Niagara Falls, Canada, July. DBLP, (2008) 379-388.
- [3] Wang P, Ji Q. Multi-view face and eye detection using discriminant features ☆[J]. Computer Vision & Image Understanding, (2007), 105(2):99-111.
- [4] Niu Z, Shan S, Yan S, et al. 2D Cascaded AdaBoost for Eye Localization, C, International Conference on Pattern Recognition. (2006)1216-1219.
- [5] Valenti R, Gevers T. Accurate eye center location through invariant isocentric patterns.[J]. Pattern Analysis & Machine Intelligence IEEE Transactions on, (2012), 34(9):1785-1798.
- [6] Asteriadis S, Nikolaidis N, Hajdu A, et al. An Eye Detection Algorithm Using Pixel to Edge Information[J]. Int.symp.on Control, 2010, september (1975), (9):761-766.
- [7] Asadifard M, Shanbezadeh J. Automatic Adaptive Center of Pupil Detection Using Face Detection and CDF Analysis[J]. Lecture Notes in Engineering & Computer Science, (2010), 21(5):13-14.
- [8] Bai L, Shen L, Wang Y. A Novel Eye Location Algorithm based on Radial Symmetry Transform[J]. (2006), 3:511-514.
- [9] Timm F, Barth E. Accurate Eye Centre Localisation by Means of Gradients., C, Visapp 2011 - Proceedings of the Sixth International Conference on Computer Vision Theory and Applications, Vilamoura, Algarve, Portugal, 5-7 March. DBLP, (2011) 125-130.
- [10] Kazemi V, Sullivan J. One millisecond face alignment with an ensemble of regression trees, C, Computer Vision and Pattern Recognition. IEEE, (2014) 1867-1874.
- [11] Smith B A, Yin Q, Feiner S K, et al. Gaze locking: passive eye contact detection for human-object interaction, C, ACM Symposium on User Interface Software and Technology. (2013) 271-280.