Eyeball Localization Based on Angular Integral Projection Function

Ghassan J. Mohammed, Bingrong Hong and Ann A. Jarjes
Computer Science and Engineering Department
Harbin Institute of Technology
Harbin, 150001, P.R. China
E-mail:ghassanjasim@yahoo.com, hongbr@hit.edu.cn, ann_kazzaz2004@yahoo.com

Ghassan J. Mohammed
Foreigner Students Building A-13
Harbin Institute of Technology
92 West Dazhi Street
Nangang District
Harbin , P. R. China
E-mail:ghassanjasim@yahoo.com

Iris boundaries localization is a critical model in facial feature-based recognition systems. It has a close relationship to the results' accuracy of these applications. In this paper, we propose a new algorithm based on the angular integral projection function (AIPF) to localize eyeball (iris outer boundary) in iris images. The proposed algorithm adopts boundary points detection and curve fitting within gray level images. Mainly it finds the iris outer boundary's features, the center and radius in three steps. First, we detect the approximate position of iris center. Then, using AIPF, a set of radial boundary points are detected. Finally, the features are obtained by fitting a circle to the detected boundary points. The experimental results on 756 iris images from CASIA show high accuracy and efficiency.

Povzetek: Razvit je nov algoritem biometričnega prepoznavanja na osnovi mrežnice.

1 Introduction

The localization of human facial features plays a very important role in biometric systems and has received a great deal of interest in recent years. Such biometric systems include iris recognition, face recognition, disease diagnosis and human-computer interaction. Iris feature is considered to be the most important feature among other facial features that include eyes, nose, mouth and eyebrows. Iris localization aims to find the parameters, the centers and radii, of the inner and outer boundaries of iris. However, the localization of eyeball or the iris outer boundary is more difficult since it is not sharp and clear as the inner boundary, and due to the occlusion caused by eyelashes and eyelids. As localization accuracy has great influence on subsequent feature extraction and classification, many different methods have been presented for the purpose of iris localization. For the first time, Daugman [4] proposed an integrodifferential operator to detect both iris inner and outer boundaries. Wildes [15] and Masek [8] used binary edge map and voting procedure, which realized by Hough transform, to detect iris boundaries. Further, Ma et al. [7] presented an iris segmentation method based on Canny edge detection and Hough transform, Tisse et al. [14] used integrodifferential operator with Hough transform, and Cui et al. [3] used Haar wavelet transform followed by

Hough transform for detecting the iris inner boundary and differential integral operator for localizing the outer boundary. More recently, other iris localization algorithms have been proposed [5] [6] [13] [18] [12]. However, most of the localization algorithms mentioned above are based on edge detection followed by Hough transform, that is to search the iris boundaries over three-parameter space exhaustively, which makes the process time-consuming ,thus they can not be employed in real time iris recognition systems. Moreover, they require threshold values to be chosen for edge detection and this may cause critical edge points being removed, resulting in failure to detect circles. In this paper, we address these two problems and propose a new algorithm to localize eyeball in iris images efficiently and accurately. As eyeball is almost circle, we localize it by extracting the features, the center and radius, of the iris outer boundary. Thus, through the paper we denote by iris features the center and radius of the iris outer boundary. The proposed algorithm adopts boundary points detection followed by curve fitting. It does not need to find all boundary points, so its localization speed is very fast.

In our earlier work [9], the AIPF has been developed as a general function to perform integral projection along angular directions, both the well known

vertical integral projection function (IPF$_v$) and horizontal integral projection function (IPF$_h$) can be viewed as special cases of AIPF. In our approach, AIPF is adopted to detect boundary points. First, the approximate position of the iris center is detected by calculating the center of mass for the binarized eye image. Then, a set of iris outer boundary's radial points are obtained using AIPF. Finally, we get the precise iris features through fitting a circle to the above boundary points by making use of the least squares method. Being evaluated on 756 eye images from CASIA [2] and quantitatively analysed based on own ground truth, the experimental results show high accuracy and efficiency. The rest of this paper is organized as follows. In Section 2, integral projection functions are described. In Section 3, the algorithm of iris features extraction is detailed. Section 4 provides the experimental results of the algorithm on CASIA iris database. Section 5 concludes the paper.

2 Projection functions

2.1 Integral projection functions

Due to their simplicity and robustly, image integral projection functions have been used widely for the detection of the boundary between different image regions. Among them, the vertical and horizontal integral projection functions are most popular. Here, suppose $I(x,y)$ is the intensity of a pixel at location (x,y), the vertical integral projection function $IPFv(x)$ and horizontal integral projection function $IPFh(y)$ of $I(x,y)$ in intervals [y1, y2] and [x1,x2] can be defined respectively as:

$$IPF_v(x) = \int_{y1}^{y2} I(x,y)dy. \qquad (1)$$

$$IPF_h(v) = \int_{x1}^{x2} I(x,y)dx. \qquad (2)$$

The above two functions are used to detect the boundary of different image regions in the vertical and horizontal directions. Assuming PF is a projection function and ξ is a small constant. Thus, if the value of PF rapidly changes form z_0 to $(z_0+\xi)$, it indicates that z_0 lie on the boundary between two homogeneous regions. In detail, given a threshold T, the vertical boundary in the image can be identified by:

$$\Theta_v = \max\left\{\left|\frac{\partial PF_v(x)}{\partial x}\right| > T\right\}. \qquad (3)$$

where Θ_v is the set of vertical boundary points, such as { $(x_1, PF_v(x_1))$, $(x_2, PF_v(x_2))$,...,$(x_k, PF_v(x_k))$}, which vertically divides the image into different areas. It is

obvious that the horizontal boundary points can be identified similarly [17].

2.2 Angular integral projection function

Besides the sets of vertical and horizontal boundary points that can be detected using IPF$_v$ and IPF$_h$ respectively. Other boundary point sets can be identified on other directions rather than those are on the vertical and horizontal directions. Considering this fact and in order to capture the boundary point sets along all directions within an image, the AIPF has been proposed in our earlier work [9] as a general function to perform integral projection along angular directions it is defined as:

$$AIPF(\theta,\rho,h) = \frac{1}{h+1}\int_{j=-h/2}^{h/2} I((x_0 + \rho\cos\theta) + (j\cos(\theta + \pi/2)),$$
$$(y_0 + \rho\sin\theta) + (j\sin(\theta + \pi/2)))d$$

$$(4)$$

where (x_0,y_0) is the image center, $I(x,y)$ is the gray level of the pixel at location (x,y), θ is the angle of the integration rectangle with x-axis, $\rho = 0,1,...,w$, w is the width of the integration rectangle, and h represents the height of the integration rectangle or the number of pixels to be integrated within each line. Specifically, the application of AIPF on θ direction carries out within an integration rectangle with $w \times h$ dimensions, it extends along a central line irradiated from the image center and having θ with x-axis. Here, it is worth to mention that even the most commonly used projection functions IPF$_v$ and IPF$_h$ can be implemented using AIPF by assigning θ=0°,180° and θ=90°,270° respectively.

3 Iris features extraction

In this section we are mainly concerned with the extraction of the iris features: iris center and iris radius in the segmented gray level eye images. Figure 1(a) shows an example of eye images used in this paper.

3.1 Approximate iris center detection

Since the centers of both iris and pupil are close to each other, we consider the pupil center as the approximate iris center. In order to determine the pupil center, the gray levels histogram is plotted and analysed. Figure 1(b) shows the histogram of gray levels for the image in Figure 1(a). Depending on the eye image histogram, a threshold value T is determined as the intensity value associated with the first important peak within histogram. Then, all intensity values in the eye image below or equal T are changed to 0(black) and above T are changed to 255(white),as:

$$g(x,y) = 255, \text{ if } I(x,y) \geq T$$
$$g(x,y) = 0, \text{ otherwise.}$$

$$(5)$$

where $I(x,y)$ is the intensity value at location (x,y), $g(x,y)$ is the converted pixel value and T represents threshold. This process converts a gray image to binary image and

efficiently segments the pupil from the rest of the image as shown in Figure 1(c). However, morphological processing is still necessary to remove pixels that located outside the pupil region. Figure 1(d) shows the clear pupil region obtained from Figure 1(c) after noise removing by using dilate operator. Now, the center of the segmented pupil can be easily determined. Basing on [1], the center of a simple object like circle and square coincides with its center of mass. The center of the mass refers to the balance point $(\bar{x}, \bar{y})$ of the object where there is equal mass in all directions:

$$\bar{x} = \frac{1}{\sum g(x,y) \in F} \sum_{g(x,y) \in F} x. \qquad (6)$$

$$\bar{y} = \frac{1}{\sum g(x,y) \in F} \sum_{g(x,y) \in F} y. \qquad (7)$$

where $g(x,y)$ is a pixel in the position (x,y), and F is the object under consideration. We make use of the equations (6) and (7) to find the pupil center $P(x_p, y_p)$ which approximates the iris center $I(x_i, y_i)$. The pupil center detection process is shown in Fig. 1
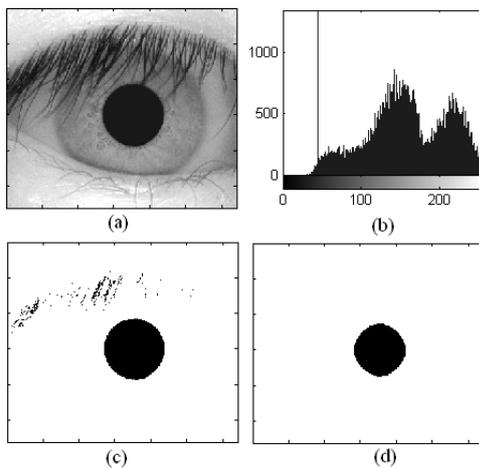
Figure 1: The pupil center detection. (a) Original image. (b)Gray levels histogram. (c) Binary image. (d) Binary image after morphological dilation operation.

3.2   Iris radius estimation

After the approximate iris center is detected, the precise iris center and radius can be estimated as the center and radius of a circle fitted to the iris outer boundary. Here, in order to reduce computational time, two rectangles are set on both sides of the iris basing on the estimated location of iris center and the number of integration rectangles that to be applied, as shown in Figure 3. In our approach, we divide the iris radius estimation task into three stages. In the first stage, as an image pre-processing step, image nonlinear filtering is performed. In the second stage, iris outer boundary points are detected. Note that both the two previous stages are

performed within the above predefined rectangles. Finally, in the third stage, a circle is fitted to these points using the least squares method. The reminder of this section details each of them.

3.2.1   Image pre-processing

As a preliminary stage to the iris boundary points detection, image filtering is performed to minimize the influence of the occlusion caused by the eyelashes in both iris' left and right rectangles, which in turns help us to detect iris boundary accurately. In this work, a nonlinear filter is adopted basing on anisotropic diffusion (an image enhancement process that removes noise and irrelevant details while preserving the edges). The filter that we used is based on the formulation of Perona and Malik [10] for the anisotropic diffusion. Applying to an image, such a filter encourages intraregional diffusion while preserving contours as shown in Figure 2. Thus, it serves to a better edge detection in a potential noisy image.
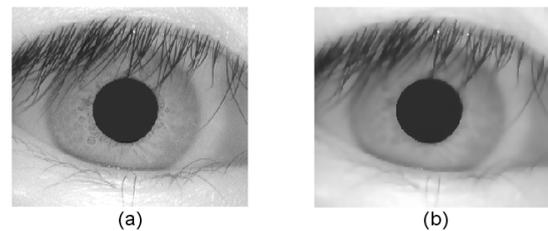
Figure 2: Anisotropic filtering based on Perona and Malik's formulation [10]. (a) Original image. (b) Filtered image.

3.2.2   Boundary points detection

From the iris image shown in Figure 1(a), it is clear that the iris is circular and darker than the surrounding area. Accordingly, considering the approximate iris center detected in the previous section as the image center, AIPF can be applied to find a set of radial boundary points. Here, in order to reduce the influence of potential occlusion caused by eyelids or eyelashes to minimum, θ is limited in the ranges -30°~ 5° and 175°~ 210° within the right iris rectangle and the left iris rectangle respectively. This is because that the iris region within these θ ranges are barely influenced by occlusion. Figure 3 shows four integration rectangles within each of the left and right iris rectangles.

Next, we find only a radial boundary point for each integration rectangle on θ direction. This is accomplished by computing the gradient of the projection curve resulted by each application of AIPF. Then we obtain the iris' boundary point by searching the gradient curve for the local maximum that corresponds to the iris edge. Clearly, the more integration rectangles thus iris' boundary points, the finer iris outer boundary localization. However, as a circle can be fit through three boundary points, at least three integration rectangles have to be established within both iris rectangles, here a

reasonable angular shift between successive integration rectangles have to be considered.

3.2.3   Curve fitting

As iris boundary is considered as a circular contour. Hence, we get the precise iris center $I(x_i, y_i)$ and radius R_i through fitting a circle to the collection of the above iris' boundary points. Figure 3 shows a circle fitted to iris boundary based on boundary point detected in previous section. Here, in order to obtain a best circle fit, we make use of the least squares method which minimizes the summed square of errors. The error for the ith boundary point r_i is defined as the difference between the detected boundary point $p\_detcted_i$ and the fitted circle point p_fitted_i, as:

$$r_i = p\_detcted_i - p\_fitted_i \qquad (8)$$

Thus the summed square of errors is given by:

$$S = \sum_{i=1}^{n} r_i^2 \qquad (9)$$

where n is the number of the detected radial boundary points.
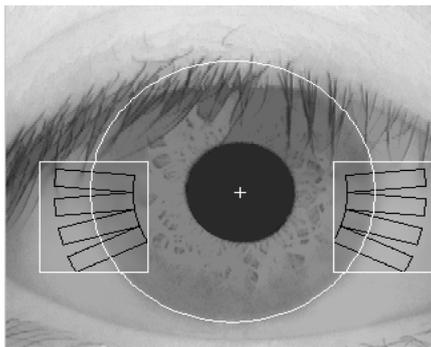
Figure 3: An example illustrates the application of the AIPF with: $\theta_1=5°$, $\theta_2=-5°$, $\theta_3=-15°$, $\theta_4=-25°$, and with $\theta_1=175°$, $\theta_2=185°$, $\theta_3=195°$, $\theta_4=205°$, in the right iris rectangle and the left integration rectangle respectively within an iris image. Each black rectangle represents an integration rectangle with $w \times h$ dimensions. The white rectangles are for the left and right iris rectangles. White cross denotes the center of the circle fitted to the iris outer boundary.

4   Experimental results

We perform experiments to evaluate the performance of the proposed algorithm over iris images supplied by CASIA [2]. The algorithm was applied for each image in the database. All experiments are performed in Matlab (version 6.5) on a PC with P4 3GHz processor and 512 M of DRAM. Figure 4 shows part of the experimental results.
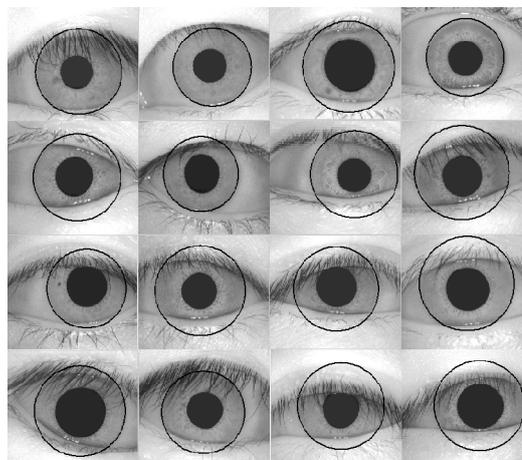
Figure 4: Eyeball localization examples. Row 1 and 2 show the localization results for typical irises and irises with different size and location. Row 3 and 4 show the localization results for irises occluded with eyelash and/or eyelid. Six integration rectangles are used with: $\theta_1=5°$, $\theta_2=0°$, $\theta_3=-5°$, $\theta_4=-10°$, $\theta_5=-15°$, $\theta_6=-20°$, in the right iris rectangle and other six are used with: $\theta_1=175°$, $\theta_2=180°$, $\theta_3=185°$, $\theta_4=190°$, $\theta_5=195°$, $\theta_6=-200°$, in the left integration rectangle. Thus 12 radial edge points are considered here. It was assumed that w=60 and h= 15.

4.1   Database characteristics

The CASIA iris database was adopted for testing. Here, it must be mentioned that this database has been manually edited [11] but it does not effect the application of AIPF to detect iris' edge points since editing was limited to the pupil region. CASIA V1.0 includes 108 classes and each class has seven iris images captured in two sessions, three in the first session and two in the second. Sessions were taken with an interval of one month. So there are totally 756 iris images with a resolution of 320×280 pixels.

4.2   Result analysis

It is known that the evaluation of the localization results of an algorithm based on observation by eye is likely to be inaccurate. Thus to achieve quantitative analysis of our algorithm's results, an analysis approach based on ground truth is adopted. In such approach, getting an accurate ground truth is vital to evaluate the performance of results. Since no ground truth is available, we hand-localized the iris center and radius for all iris images within the database, they serve as ground truth. Here, we localize the iris center and radius by manually fitting a circle on the iris outer boundary through three steps, all by hand. First, we identify the approximate iris center location, here we can show the location of the calculated iris center as a reference point. Then, we estimate the iris radius by finding a point located on the iris outer boundary. Finally, considering the center and radius

obtained from the previous two steps, a moveable and resizable circle is fitted to the iris outer boundary. After that and according to [16], a circularity confidence interval centered at the hand-localized pixel with five pixel radius, is defined. Let $H(i,j)$ denote hand-localized pixel and $E(i,j)$ the detected pixel. The distance Dis of H and E is defined as $\|H\text{-}E\|_2$. Then the accuracy of the algorithm is:

$$A = \begin{cases} (1 - Dis/5 \times 0.5) \times 100\%, & Dis \le 5 \\ 0, & Dis > 5 \end{cases} \qquad (10)$$

The satisfactory factor is set to 0.5 in Eq.(10). That is to say, the accuracy is 50% if the detected pixel position is on the boundary of the confidence interval. If it is out of the confidence interval, the accuracy is set to 0. Meanwhile, the accuracy is 100% if the detected pixel and hand-localized one are at the same position. For more reliable performance evaluation, Daugman's integrodifferential method [4] as a prevailing segmentation method, is also implemented on the same database. The experiments were done under the condition that the same initial estimate of iris center is provided for both AIPF and Daugman methods. For the application of AIPF, six integration rectangles were adopted with: (θ_1=5°, θ_2=0°, θ_3=-5°, θ_4=-10°, θ_5=-15°,θ_6=-20°) in the right iris rectangle, other six integration rectangles are adopted with: (θ_1=175°, θ_2=180°, θ_3=185°, θ_4=190°, θ_5=195°, θ_6=-200°) in the left integration rectangle, and each integration rectangle has 60×15 dimensions. It is clear form Table 1 that AIPF method achieves better accuracy than that of Daugman since 72.75 % of the AIPF's detected iris centers are within the 5-pixel confidence interval of the ground truth, and 89.4% of the AIPF's estimated iris radii are within the confidence interval. Moreover, AIPF method performs faster than Daugman method. Therefore, the proposed method demonstrates high accuracy with faster execution.

Method	Accuracy within 5- pixel confidence		Time		
	Iris center	Iris radius	Mean	Min.	Max.
Daugman	69.84 %	79.1 %	0.73s	0.36s	1s
Proposed	72.75 %	89.4 %	0.27s	0.23s	0.4s

Table 1. Performance of the AIPF method.

5   Conclusion

An algorithm for the localization of eyeball or the iris outer boundary is reported in the paper. The proposed algorithm adopts radial edges detection with curve fitting in gray level iris images. First, the rough iris center is detected. Then, a set of radial edge points are detected based on AIPF. Finally, getting the precise iris features through fitting a circle to the detected edge points. Experimental results on a set of 756 iris images from CASIA V1.0 indicate high accuracy and faster execution due to its simplicity in implementation. Our future work has two fields. First, we will perform iris segmentation based on AIPF for iris recognition. Second, we will utilize AIPF for detecting other facial features.

References

[1] Baxes, G. A.: Digital Image Processing: Principles and Applications, Wiley, New York, 1994.

[2] Chinese Academy of Sciences - Institute of Automation, CASIA Iris Image Database (ver. 1.0), Available on: http://www.sinobiometrics.com.

[3] Cui, J., Wang, Y., Tan, T., Ma, L., Sun, Z.: An Iris Recognition Algorithm Using Local Extreme Points, Proceedings of the First International Conference on Biometrics Authentication, ICBA '04, Hong Kong, 2004, pp. 442-449.

[4] Daugman, J. G.: High Confidence Visual Recognition of Persons by a Test of Statistical Independence, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 15, 1993, No. 11, pp. 1148–1161.

[5] Feng, X., Fang, C., Ding, X., Wu, Y.: Iris Localization with Dual Coarse-to-fine Strategy, Proceedings of the 18th International Conference on Pattern Recognition, ICPR'06, 2006, pp. 553-556.

[6] He, X., Shi, P.: A Novel Iris Segmentation Method for Hand-held Capture Device, Proceedings of the International Conference on Biometrics, ICB 06, Hong Kong, China, 2006, pp. 479-485.

[7] Ma, L., Tan, T., Wang, Y., Zhang, D.: Efficient Iris Recognition by Characterizing Key Local Variations, IEEE Transactions on Image Processing, Vol. 13, 2004, pp. 739-750.

[8] Masek, L. : Recognition of Human Iris Patterns for Biometric Identification, Master Thesis, University of Western Australia, 2003.

[9] Mohammed, G., Hong, B., Al-Kazzaz, A.: Accurate Pupil Features Extraction Based on New Projection Function, Computing and Informatics, To be appear in Vol. 29, 2010.

[10] Perona, P., Malik, J.: Scale-Space and Edge Detection Using Anisotropic Diffusion, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 12, 1990, No.7, pp. 629-639.

[11] Philips, P., Bowyer, K., Flynn, P.: Comments on The CASIA Version 1.0 Iris Data Set, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 29, 2007, No. 10.

[12] Sudha, N., Puhan, N., Xia, H., Jiang, X.: Iris Recognition on Edge Maps, IET Computer Vision , Vol. 3, 2009, No.1, pp. 1-7.

[13] Sun, C., Zhou, C., Liang, Y., Liu, X. : Study and Improvement of Iris Location Algorithm, Proceedings of the International Conference on Biometrics, ICB 06, Hong Kong, China, 2006, pp. 436-442.

[14] Tisse, C., Martin, L., Torres, L., Robert, M.: Person Identification Technique Using Human Iris Recognition, Proceedings of the 15th International Conference on Vision Interface, VI '02, Calgary, Canada, 2002, pp. 294-299.

[15] Wildes, R. P.: Iris Recognition: An Emerging Biometric Technology, Proceedings of the IEEE, Vol. 85, 1997, No. 9, pp. 1348–1363.

[16] Zheng, Z., Yang, J., Yang, L.: A Robust Method for Eye Features Extraction on Color Image, Pattern Recognition Letters, Vol. 26**,** 2005, No. 14, pp. 2252-2261.

[17] Zhou, Z., Geng, X.: Projection Functions for Eye Detection, Pattern Recognition, Vol. 37, 2004, pp. 1049-1056.

[18] Zuo, J., Ratha, N. , Connell, J.: A New Approach for Iris Segmentation, Proceeding of IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Anchorage, Alaska, 2008,pp. 1-6.