Computing and Informatics, Vol. 29, 2010, 663–680

ACCURATE PUPIL FEATURES EXTRACTION BASED ON NEW PROJECTION FUNCTION

Ghassan J. Mohammed, Bing-Rong Hong, Ann A. Jarjes

Computer Science and Engineering Department
Harbin Institute of Technology
92 West Dazhi Street, Nangang District
Harbin, 150001, P. R. China
e-mail: {ghassanjasim, ann_kazzaz2004}@yahoo.com, hongbr@hit.edu.cn

Manuscript received 4 June 2008; revised 16 April 2009
Communicated by Steve J. Maybank

Abstract. Accurate pupil features extraction is a key step for iris recognition. In this paper, we propose a new algorithm to extract pupil features precisely within gray level iris images. The angular integral projection function (AIPF) is developed as a general function to perform integral projection along angular directions, both the well known vertical and horizontal integral projection functions can be viewed as special cases of AIPF. Another implementation for AIPF based on localized Radon transform is also presented. First, the approximate position of pupil center is detected. Then, a set of pupil's radial boundary points are detected using AIPF. Finally, a circle to the detected boundary points is fitted. Experimental results on 2 655 iris images from CASIA V3.0 show high accuracy with rapid execution time.

Keywords: Pupil features extraction, human facial features, biometrics, integral projection functions, localized Radon transform

1 INTRODUCTION

In the recent years, human facial features extraction has played important role for a wide range of usage, such as facial expression analysis, feature-based personal recognition and human-computer interaction. Such facial features include eyes, nose, mouth and eyebrows and so on. Among them, eyes are the most important salient

feature providing crucial information for personal identification based on facial features. Commonly, the iris and the pupil are the most important eye features. Many applications require accurate pupil features (pupil center and radius) detection such as iris recognition, gaze detection and disease diagnosis. As an example, Proença and Alexandre [9] observed a significant degradation of iris recognition rates, especially in the presence of translation errors of the segmented pupil border. Here, error values higher than 1 pixel significantly increased the false rejection rates. In addition to the classical eye features localization approaches (Daugman [5] and Wildes [12]), many other algorithms have been proposed in recent years in [11, 8, 7, 4, 6]. However, accurate pupil features detection is still a challenging problem. Besides, most of the above methods are based on edge detection and finding the pupil and iris boundaries upon using circular edge detector or Hough transform, which involves two drawbacks. First, the quantity of data needed to calculate is very large resulting in low speed. Secondly, 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 propose a new algorithm for accurate pupil features extraction in iris images, in which we adopt boundary points detection followed by curve fitting. It does not need to find all pupil's boundary points, so its localization speed is fast. As a general function to perform integral projection along angular directions, the angular integral projection function (AIPF) has been developed; both the well known vertical integral projection function (IPFv) and horizontal integral projection function (IPFh) can be viewed as special cases of AIPF. Also, another implementation for AIPF based on localized Radon transform is presented. First, the approximate position of the pupil center is detected by calculating the center of mass for the binarized eye image. Then, a set of pupil's radial boundary points are obtained using AIPF. Finally, we get the precise pupil features through fitting a circle to the above boundary points by making use of the least squares method. Due to the lack of ground truth [13], the algorithm's performance is evaluated based on our own implemented ground truth. Also, the selection of the method parameters is investigated in detail. Being evaluated on 2 655 eye images from CASIA V3.0 [2], the experimental results show high accuracy with rapid execution time.

The rest of this paper is organized as follows. In Section 2, the integral projection functions and the proposed AIPF are presented. Section 3 describes the details of the pupil features extraction algorithm. Section 4 gives 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 robustness, 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 $\mathrm{IPF}_v(x)$ and horizontal integral projection function $\mathrm{IPF}_h(y)$ of $I(x, y)$ in intervals $[y_1, y_2]$ and $[x_1, x_2]$ can be defined as follows:

$$\mathrm{IPF}_v(x) = \int_{y_1}^{y_2} I(x, y)\,\mathrm{d}y, \tag{1}$$

$$\mathrm{IPF}_h(y) = \int_{x_1}^{x_2} I(x, y)\,\mathrm{d}x. \tag{2}$$

The above two functions are used to detect the boundary of different image regions in the vertical and horizontal directions. Let us assume 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 may lie on the boundary between two homogeneous regions. In detail, given a threshold T, the vertical boundary in the image can be identified by:

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

where Θ_v is the set of vertical boundary points, such as $\{(x_1, PF_v(x_1)), (x_2, PF_v(x_2)), \ldots, (x_k, PF_v(x_k))\}$, which vertically divides the image into different areas. It is obvious that the horizontal boundary points can be identified similarly [14].

2.2 Angular Integral Projection Function

Besides the sets of vertical and horizontal boundary points that can be detected using IPF_v and IPFh, respectively. It is worth noticing that other boundary point sets can be identified on other directions rather than those on the vertical and horizontal directions. This could be realized by the application of the integral projection along angular directions, each within a specified area named the integration rectangle, since performing line integrals within a specified area results in more accurate boundary points detection. Owing to the above and in order to capture the boundary point sets for all directions within an image, we proposed a new method, namely angular integral projection function (AIPF). AIPF performs line integration within a specified rectangle, that extends from the image center and having θ angle with x-axis, it is defined as follows:

$$\mathrm{AIPF}(\theta, \rho, h) = \frac{1}{h+1}\int_{j=-\frac{h}{2}}^{\frac{h}{2}} I((x_0 + \rho\cos\theta) + (j\cos(\theta + \pi/2)), (y_0 + \rho\sin\theta) + (j\sin(\theta + \pi/2)))\,\mathrm{d}j, \tag{4}$$

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, $\rho = 0, 1, \ldots, w$, 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. Therefore, the application of AIPF on θ direction carries out within an integration rectangle with $w \times h$ dimensions that extends along a central line emanated from the image center

having θ with x-axis. Figure 1 explains the approach with multiple integration rectangles. Here, it is worth to mention that even the most commonly used projection functions IPFv and IPFh can be implemented using AIPF by assigning $\theta = 0, 180°$ and $\theta = 90, 270°$, respectively.

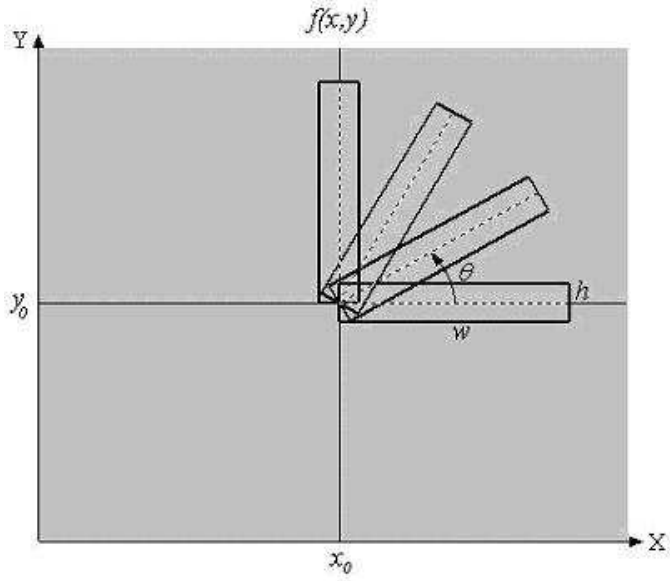


Fig. 1. An example illustrates the application of the proposed AIPF with four angles as: $\theta_1 = 0°, \theta_2 = 30°, \theta_3 = 60°, \theta_4 = 90°$ and $\Delta\theta = 30°$, within an image $f(x, y)$ centered at (x_0, y_0). Each rectangle represents an integration rectangle with $w \times h$ dimensions. The dotted lines denote the integration rectangles' central lines.

2.3 Implementing AIPF Based on Localized Radon Transform

Radon transform is actually a process of line integral, integrating pixel values along all possible lines in an image space. Accordingly, another implementation of AIPF can be obtained based on localized Radon transform. Radon transform over two-dimensional Euclidean space can be defined as follows [10]:

$$f(\theta, \rho) = \Re\{I\} = \int \int_D I(x, y)\delta(\rho - x\cos\theta - y\sin\theta)\,\mathrm{d}y\mathrm{d}x, \tag{5}$$

where

- D is the x-y image plane,
- $I(x, y)$ is the image intensity at position (x, y),
- δ is the Dirac delta function,

- ρ is the length of the normal from origin to the straight line,
- θ is the angle between the normal and the x-axis.

In contrast to Radon transform, localized Radon transform (LRT) performs line integration in a small rectangle that can be shifted throughout the image. Copeland [3] implemented the localized Radon transform as the original transform (5) with limits on the integration, and with an additional parameter that controls shifting between adjacent rectangles. In our case, we present a new approach for implementing LRT by setting new limits on the original transform (5); we define the proposed LRT as

$$f(\theta,\rho) = \Re_{Loc}\{I\} = \int_{X_{min}}^{X_{max}} \int_{Y_{min}}^{Y_{max}} I(x,y)\delta(\rho - x\cos\theta - y\sin\theta)\,\mathrm{d}x\mathrm{d}y, \qquad (6)$$

where

- $X_{min} = \min((h/2)\cos(\theta+\pi/2) + \rho\cos(\theta), (h/2)\cos(\theta+\pi/2) + \rho\cos(\theta))$,
- $X_{max} = \max((h/2)\cos(\theta+\pi/2) + \rho\cos(\theta), (h/2)\cos(\theta+\pi/2) + \rho\cos(\theta))$,
- $Y_{min} = \min((h/2)\sin(\theta+\pi/2) + \rho\sin(\theta), (h/2)\sin(\theta+\pi/2) + \rho\sin(\theta))$,
- $Y_{max} = \max((h/2)\sin(\theta+\pi/2) + \rho\sin(\theta), (h/2)\sin(\theta+\pi/2) + \rho\sin(\theta))$,
- ρ is the length of the normal from the origin to the straight line, $\rho = 0, 1, 2, \ldots, w$,
- h represents the length of line parameterized by (ρ, θ), (number of points to be integrated).

Notice that to perform the line integration along the whole integration rectangle under consideration the proposed LRT has to be applied for each value of ρ, as $\rho = 0, 1, 2, \ldots, w$.

3 PUPIL FEATURES EXTRACTION

This section is mainly concerned with extraction of the pupil features: pupil center and pupil radius in the segmented gray level eye images. Figure 2 a) shows an example of eye images used in this paper.

3.1 Image Filtering and Enhancement

As in the iris image shown in Figure 2 a), the pupil region has eight specular reflection spots caused by the NIR illuminators used to collect the iris images of CASIA database. Since these reflection spots cause false pupil boundary detection, a technique is required to remove these artifacts. Here, we apply simple technique to remove these spots. First, the complement of the image is evaluated by calculating the absolute subtraction of each pixel's gray level from 255, then filling the resulting dark holes in the intensity image, where a hole is the set of dark pixels surrounded

by light pixels and can not be reached from the edge of the image. Here, the connectivity of 4-connected pixels is adopted on the background pixels. Finally the complement of the processed image is evaluated again. However, this technique results in sharp image, so Gaussian smoothing is followed. Figure 2 illustrates the approach.

Although the above approach is proved valid, it is not optimal. Some residue may still be left after removing the reflection spots. However, this is not a major problem and has limited effect on the later boundary points detection, since the contrast between the pupil and residue regions is still low as compared to the high contrast between pupil and iris. In most cases, the above approach does well in removing the reflection spots except for the situation when these spots are located on the pupil edge and are connected with iris region. Figure 3 shows some examples where pupil boundary points can not be located correctly due to reflection spots, which of course degrades the accuracy of the detected pupil features.

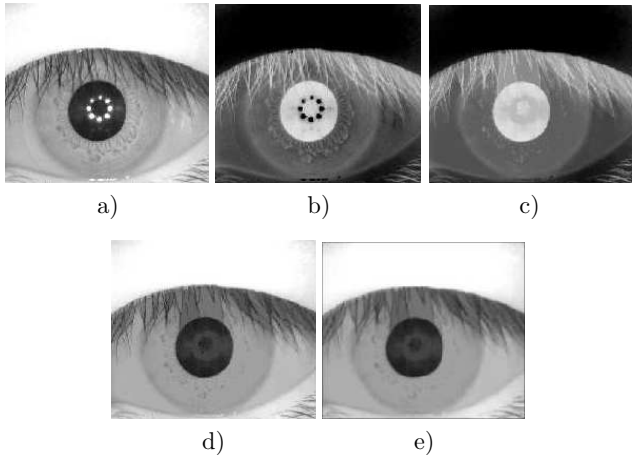


Fig. 2. Iris image preprocessing and enhancement: a) Original image, b) Complement of image, c) Filling the holes, d) Complement of the image in c), e) Resulting image after Gaussian smoothing

3.2 Approximate Pupil Center Detection

In order to determine the pupil center, the gray levels histogram is plotted and analyzed. Here, we only consider a middle sub-image that is around 1/2 part of the iris image, since most pupils are located near the center of the iris image. This makes the process more efficient and reduces the effect of potential occlusion of the dark areas caused by eyelashes and low illumination near image border. Figure 4 c) shows the histogram of gray levels for the sub-image in Figure 4 b). Depending on histogram, a threshold value T is determined as the intensity value associated with

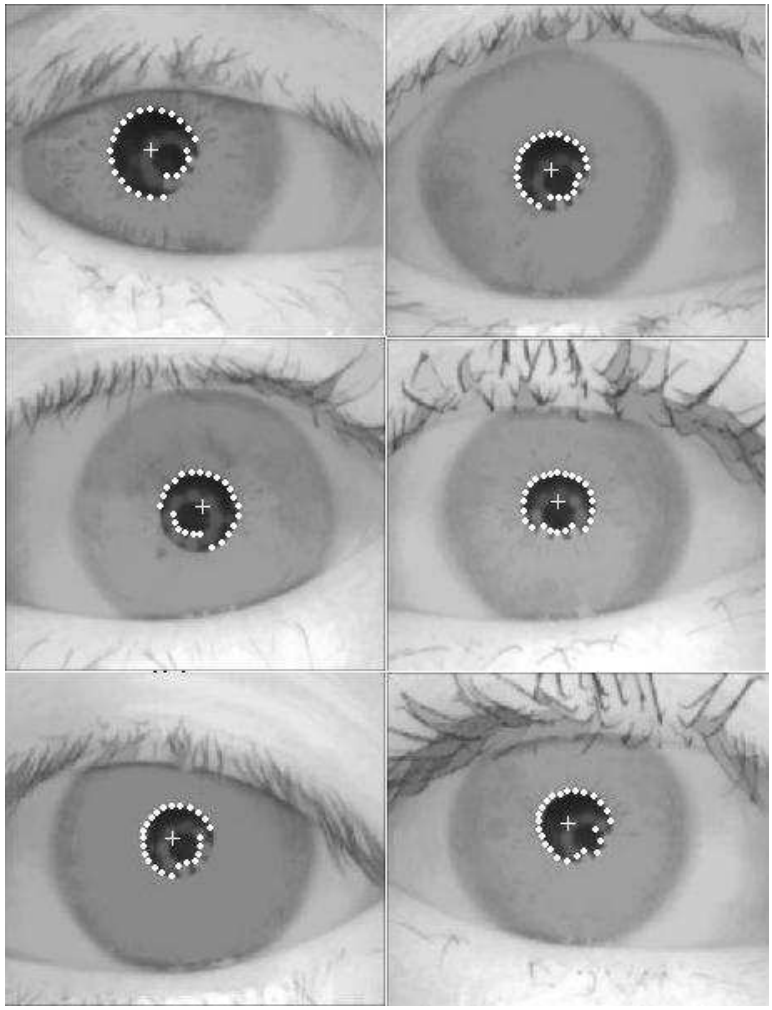


Fig. 3. Pupil boundary points localization where the reflection spots are located on the pupil edge. The white cross represents approximate pupil center.

the first important peak within the histogram. Then, all intensity values in the eye image below or equal to T are changed to 0 (black) and above T are changed to 255 (white), as:

$$\begin{aligned} g(x,y) &= \quad 255, \text{ if } I(x,y) \geq T \\ g(x,y) &= \qquad 0, \text{ otherwise,} \end{aligned} \tag{7}$$

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 in Figure 4 d)). However, morphological processing is still necessary to remove pixels located outside the pupil region. Figure 4 e) shows the clear pupil region obtained from Figure 4 d) after noise removing by using dilate operator. Now, the center of the segmented pupil can be easily determined. Based on [1], the center of a simple object like circle and square coincides with its center of mass. The center of mass refers to the balance point $(\bar{x}, \bar{y})$ of the object where there is equal mass in all directions:

$$\bar{x} = \frac{1}{\sum g(x, y) \in F} \sum_{g(x,y) \in F} x, \tag{8}$$

$$\bar{y} = \frac{1}{\sum g(x, y) \in F} \sum_{g(x,y) \in F} y, \tag{9}$$

where $g(x, y)$ is a pixel in the position (x, y), and F is the object under consideration. We make use of Equations (8) and (9) to find the approximate pupil center (x_p, y_p).

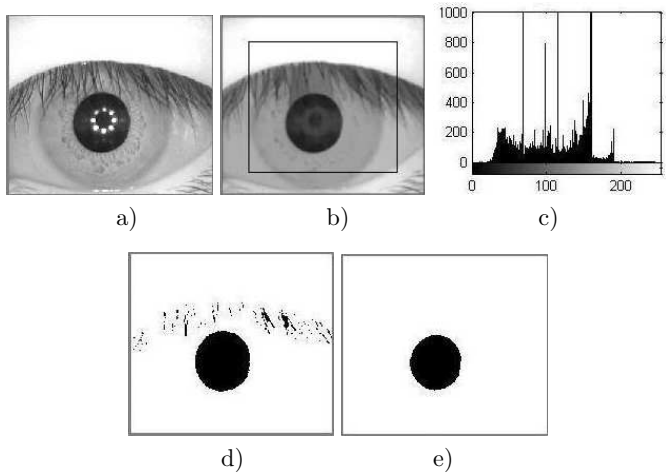


Fig. 4. Approximate pupil center detection: a) Original image, b) Processed image, c) Gray levels histogram for the sub-image within the black rectangle in b), d) Binary image, e) Binary image after morphological dilation operation

3.3 Pupil Radius Estimation

Once the approximate pupil center is detected, the precise pupil center and radius can be estimated as the center and radius of a circle fitted to the pupil boundary. We

achieve pupil boundary localization in two stages. In the first stage, pupil boundary points are detected. In the second stage, a circle is fitted to these points using the least squares method. The reminder of this section details each of them.

3.3.1 Boundary Points Detection

As common, image integral projection functions are used widely for the detection of the boundary points between different image regions. Accordingly, the proposed AIPF can be applied to find a set of boundary points since the intensity of eye pupil is different (lower) from surrounding area, as shown in Figure 2 a). Here the approximate pupil center detected in previous section is considered as the image center, and θ runs in $[0, 2\pi]$. As in AIPF's definition, each application of AIPF on specified θ direction establishes an integration rectangle having θ with x-axis. Based on the assumption that the pupil boundary has circular contour, the accuracy of later circle fitting for the pupil boundary can be improved by selecting all integration rectangles (thus boundary points) being equally spaced around the whole circumference of the pupil. To ensure that, we compute the values of θ required for the successive applications of AIPF as:

$$\theta_{next} = \theta_{current} + \triangle\theta, \tag{10}$$

where θ_{next} represents the value of θ to be considered for the next application of AIPF, $\theta_{current}$ represents θ for the current application, and $\triangle\theta$ denotes the angular spacing step. Since θ varies within $[0, 2\pi]$, therefore the first integration rectangle has $\theta = 0^\circ$. Figure 5 illustrates the approach.

Next, we find a radial boundary point for each integration rectangle, with one boundary point on each θ direction. This is accomplished by computing the gradient of the projection curve resulting from each application of AIPF. Then a pupil boundary point is obtained by searching the projection gradient curve for the local maximum that corresponds to the pupil edge. Clearly, the more integration rectangles thus pupil's boundary points, the finer pupil boundary localization.

3.3.2 Curve Fitting

As mentioned above, the pupil boundary is considered as a circular contour. Hence, we get the precise pupil center $I(x_p, y_p)$ and radius R_p through fitting a circle to the collection of the above pupil's boundary points. 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 i^{th} 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

$$r_i = p\_detcted_i - p\_fitted_i. \tag{11}$$

Thus the summed square of errors is given by

$$S = \sum_{i=1}^{n} r_i^2 , \tag{12}$$

where n is the number of the detected radial boundary points.

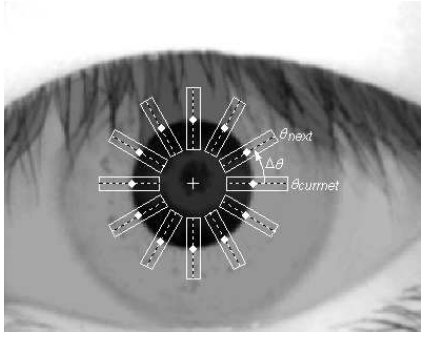


Fig. 5. The application of AIPF for finding 12 pupil boundary points (white diamonds), for $\Delta\theta = 30°$ and $\theta = 0°(\theta_{current})$ for the first integration rectangle. Each rectangle denotes the integration rectangle resulting form each application of AIPF with different θ. White cross represents the approximate pupil center.

4 EXPERMENTAL RESULTS

CASIA iris database V3.0 [2] was adopted in our experiments. Comprising 2 655 iris images, CASIA V3.0 is considered as the largest iris database that can be found in public domain, thus providing good and reliable environment to evaluate the performance of our algorithm. The algorithm was applied for each image in the database. All experiments are performed in Matlab (version 6.5) on a PC with P4 3 GHz processor and 512 M of DRAM. Figure 6 shows nine samples of pupil boundary localization.

4.1 Database Characteristics

The adopted CASIA iris database includes 2 655 iris images from 249 subjects, with 396 total number of iris classes. Most of the images were captured in two sessions with at least one month interval. The iris images are 8-bit gray level images with a resolution of 320×280 pixels.

4.2 Result Analysis

A common approach to achieve quantitative analysis for an algorithm's result is that involving building and using ground truth. As such approach is adopted to

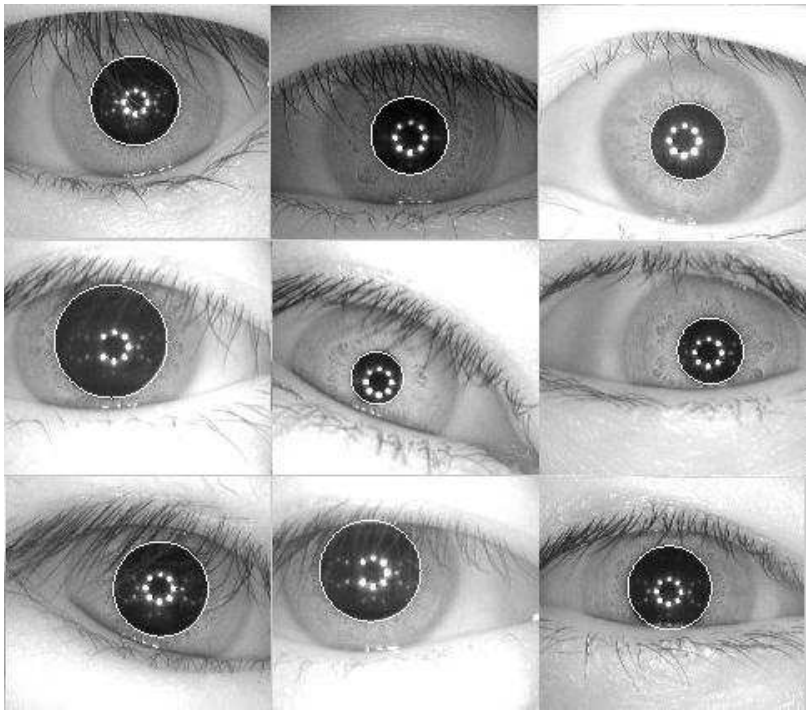


Fig. 6. Pupil boundary localization examples. Row 1 shows the localization results for typical pupils. Row 2 shows the localization results for different pupil sizes and locations. Row 3 shows the localization results for pupils occluded with eyelashes and/or eyelids. It was assumed that $\Delta\theta = 15°$, thus 24 radial edge points are considered here, with $w = 80$ pixels and $h = 15$ pixels.

analyze our algorithm's results, so getting accurate ground truth is essential due to the lack of precise ground truth. For our case, and in order to generate ground truth, we consider the pupil center and radius for the manually fitted circle on the pupil boundary through three steps, all by hand. First, we identify the approximate pupil center location; here we can show the location of the calculated pupil center as a reference point. Then, we estimate the pupil radius by finding a point located on the pupil boundary. Finally, considering the center and radius obtained from the previous two steps, a moveable and resizable circle is fitted to the pupil boundary.

After that and according to [13], 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 $\|H - E\|_2$. Then the accuracy of the algorithm is:

$$A = \begin{cases} (1 - Dis/5 \times 0.5) \times 100\,\%, & Dis \leq 5 \\ 0, & Dis > 5. \end{cases} \tag{13}$$

The satisfactory factor is set to 0.5 in Equation (13). 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 and hand-localized pixels are at the same position.

	h	Accuracy within 1-pixel confidence (%)		Times(s)		
$\Delta\theta$	(pixels)	Pupil center	Pupil radius	Mean	Min.	Max.
	11	87.35	92.73	0.26	0.23	0.34
	13	87.5	92.84	0.27	0.23	0.33
	15	87.68	92.35	0.27	0.25	0.34
	17	88.14	92.02	0.27	0.25	0.38
10°	19	87.76	90.62	0.28	0.25	0.38
	21	87.83	88.25	0.28	0.25	0.33
	23	87.8	84.86	0.29	0.27	0.38
	25	88.02	82.07	0.29	0.27	0.44
	11	86.33	92.81	0.24	0.22	0.31
	13	86.44	92.96	0.25	0.22	0.34
	15	87.12	93.37	0.25	0.22	0.36
	17	87.19	92.92	0.25	0.22	0.34
15°	19	86.52	92.54	0.25	0.22	0.34
	21	86.4	91.56	0.26	0.23	0.38
	23	86.74	89.68	0.26	0.23	0.34
	25	87.01	87.23	0.26	0.23	0.34
	11	83.32	91.11	0.23	0.2	0.34
	13	84.07	91.6	0.23	0.2	0.33
	15	84.37	91.64	0.23	0.2	0.3
	17	84.93	91.86	0.23	0.2	0.3
30°	19	84.11	90.96	0.23	0.2	0.34
	21	84.22	89.79	0.23	0.2	0.31
	23	84.11	87.76	0.23	0.2	0.33
	25	83.99	85.65	0.23	0.2	0.3
	11	82.67	91.75	0.22	0.19	0.33
	13	83.43	91.49	0.22	0.19	0.28
	15	85.24	91.79	0.22	0.19	0.31
	17	85.05	91.04	0.22	0.2	0.31
45°	19	84.29	89.76	0.22	0.19	0.31
	21	84.82	88.44	0.22	0.2	0.31
	23	84.52	86.1	0.22	0.19	0.34
	25	84.3	82.64	0.22	0.2	0.31

Table 1. Pupil features extraction accuracy and time for different $\Delta\theta$ and h parameters

The parameters w, h and $\triangle\theta$ control the number and location of pixels within each integration rectangle. Thus the selection of these parameters has important effect on both the results accuracy and the computational time of the proposed algorithm. For our experiments the value of w was set to 80 as the maximum

Method	Accuracy within 1-pixel confidence (%)		Times(s)		
	Pupil center	Pupil radius	Mean	Min.	Max.
Daugman	75.7	78.4	0.88	0.71	1.23
Proposed	87.1	93.4	0.25	0.22	0.36

Table 2. Performance of the AIPF method

expected pupil radius length for the iris image database based on earlier knowledge. To demonstrate the change in performance obtained by choosing different values for the other parameters h and $\triangle\theta$, and thus to obtain the best parameters, the proposed algorithm was applied for $\triangle\theta = 10°, 15°, 30°$ and $45°$, and each for $h = 11, 13, 15, 17, 19, 21, 23$ and 25 pixels; thus it has been applied for different 32 parameters. The results are given in Figures 7 and 8 and in Table 1. As the

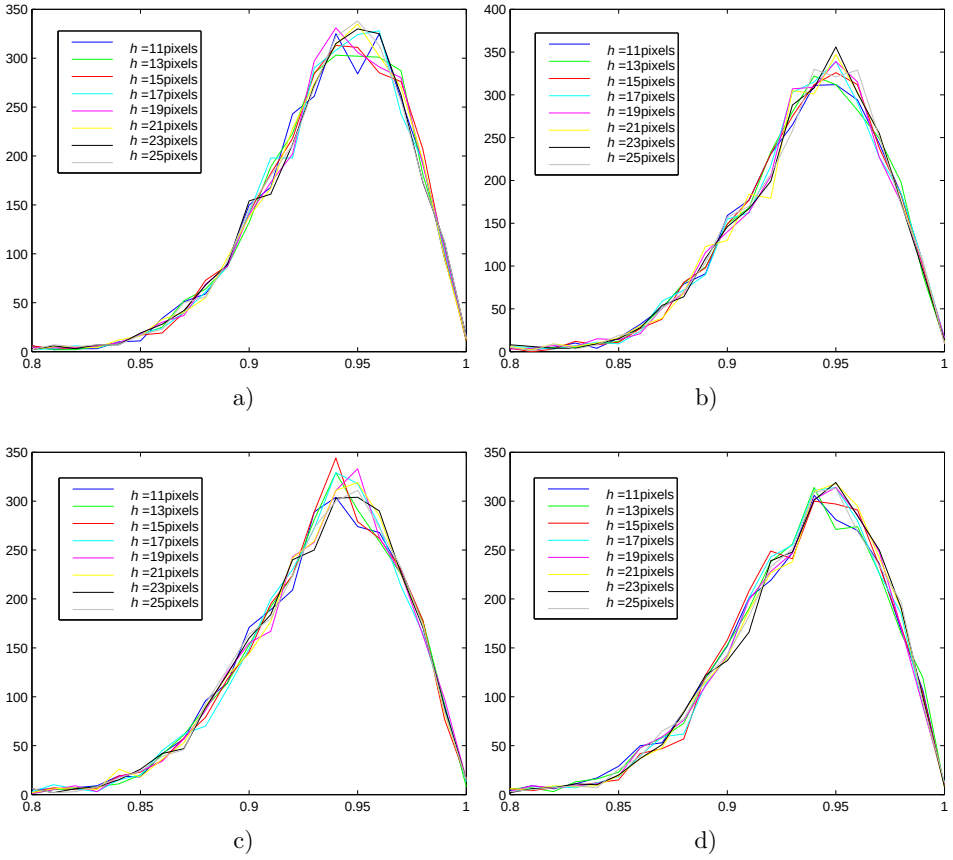


Fig. 7. Pupil center extraction accuracy at different $\Delta\theta$ values, each for $h = 11, 13, 15, 17, 19, 21, 23$ and 25 pixels: a) $\Delta\theta = 10°$, b) $\Delta\theta = 15°$, c) $\Delta\theta = 30°$, d) $\Delta\theta = 45°$

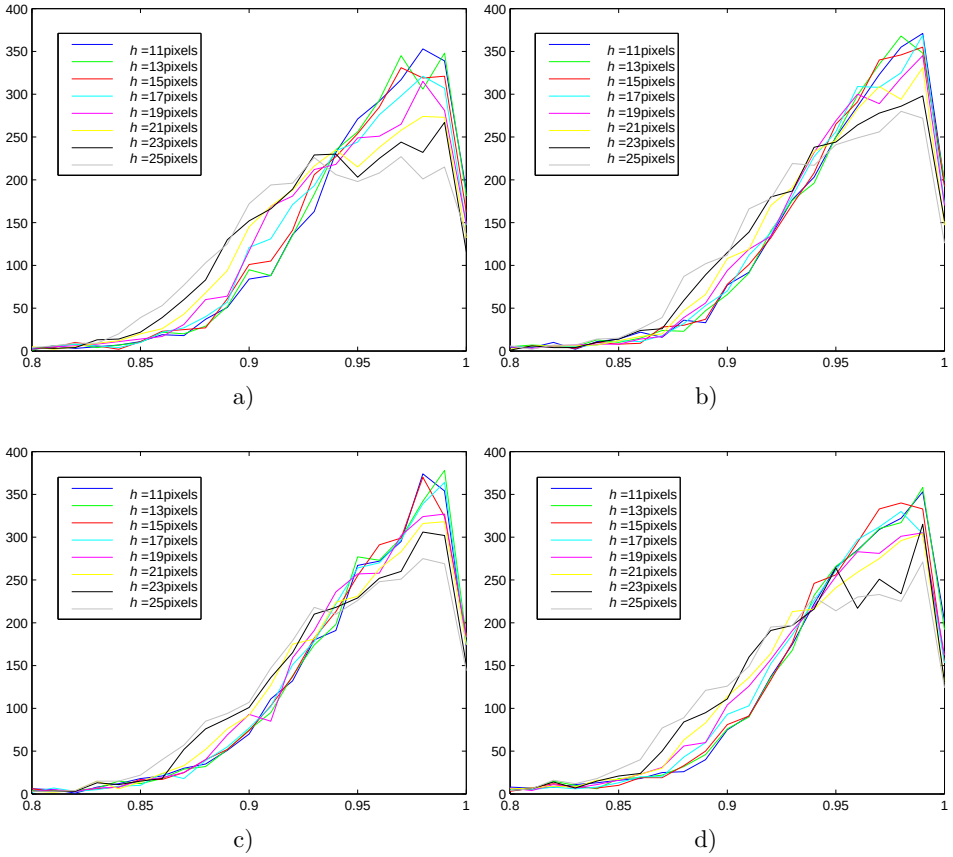
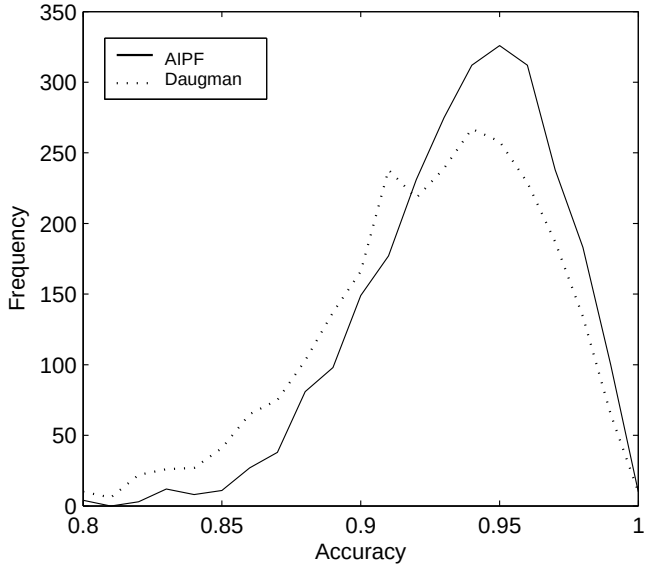


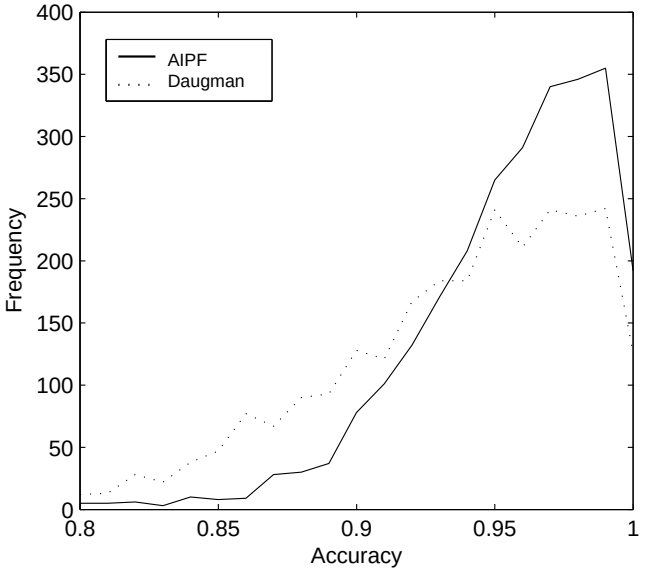
Fig. 8. Pupil radius extraction accuracy at different $\Delta\theta$ values, each for $h = 11, 13, 15, 17, 19, 21, 23$ and 25 pixels: a) $\Delta\theta = 10°$, b) $\Delta\theta = 15°$, c) $\Delta\theta = 30°$, d) $\Delta\theta = 45°$

results indicate, setting $\triangle\theta$ at small value increases the number of detected edge points and thus achieves high accurate results, but consumes more execution time, whereas setting $\triangle\theta$ at large value decreases the number of detected edge points which results in relatively less accurate results, but consumes less execution time. Therefore, a tradeoff is required between accuracy and speed. It can also be seen that, as the value of h increases, the number of calculated pixels and execution time will increase accordingly. For our results analysis, the performance differences are not very significant in general when changing the values of h and $\triangle\theta$; however, setting h at either 15 or 17 pixels produces better results.

For more reliable performance evaluation, Daugman's integrodifferential method [5] as a prevailing segmentation method has also been implemented on the same database. The experiments were done under the condition that the same initial estimate of pupil center is provided for both AIPF and Daugman methods.



a)



b)

Fig. 9. Pupil features extraction performance, considering that $\Delta\theta = 15°$, $w = 80$ pixels and $h = 15$ pixels, a) Pupil center, b) Pupil radius

The comparisons of detection performance between AIPF for $\triangle\theta = 15^\circ$ and $h = 15$ pixels as the best obtained parameters, and Daugman method for both pupil center and radius are shown in Figure 9 a) and Figure 9 b), respectively.

It is clear form Table 2 that AIPF method achieves better accuracy than that of Daugman since 87.1 % of the AIPF's detected pupil centers are within 1-pixel confidence interval of the ground truth, and 93.4 % of the AIPF's estimated pupil radii are within 1-pixel confidence interval. Moreover, AIPF method performs faster than Daugman method. Therefore, the proposed method demonstrates high accuracy with faster execution.

5 CONCLUSION

An algorithm for accurate pupil center and radius extraction is reported in the paper. The angular integral projection function (AIPF) is developed. Another implementation of AIPF based on localized Radon transform is also proposed. The proposed algorithm adopts radial edges detection with curve fitting in gray level iris images. First, the rough pupil center is detected. Then, a set of radial edge points are detected based on AIPF. Finally, getting the precise pupil features through a circle fitting to the detected edge points. The selection of the method parameters has also been investigated in detail. Experimental results on a set of 2 655 iris images from CASIA V3.0 indicate high accuracy and faster execution due to its simplicity in implementation. Our future work has two fields: first, focus on the extension of AIPF to other projection functions, and second, involving utilizing AIPF for the detection of 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. 3.0). Available on: `http://www.sinobiometrics.com`.

[3] Copeland, A. C.—Ravichandran, G.—Trivedi, M. M.: Localized Radon Transform-Based Detection of Ship Wakes in SAR Images. IEEE Transactions on Geoscience and Remote Sensing, Vol. 33, 1995, No. 1, pp. 35–45.

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

[5] 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.

[6] Dey, W.—Samanta, D.: An Efficient Approach for Pupil Detection in Iris Images. In Proceedings the IEEE 15th International Conference on Advanced Computing and Communications Recognition, ADCOM 2007, Bangalore, India, 2007, pp. 382–387.

[7] Ma, L.—Tan, T.—Wang, Y.—Zhang, D.: Personal Recognition Based on Iris Texture Analysis. IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 25, 2003, pp. 1519–1533.

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

[9] Proença, H.—Alexandre, L. A.: A Method for the Identification of Inaccuracies in Pupil Segmentation. In Proceedings of the IEEE First International Conference on Availability, Reliability and Security, ARES '06, Vienna, Austria, 2006, pp. 224–228.

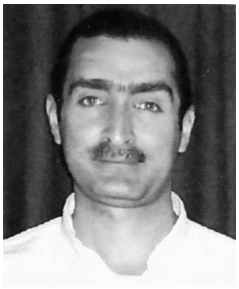
[10] Rey, M. T.—Tunaley, J. K. E.—Folinsbee, J. T.—Jahans, P. A.—Dixon, J. A.—Vant, M. R.: Application of Radon Transform Techniques to Wake Detection in Seasat-A SAR Images. IEEE Transactions on Geoscience and Remote Sensing, Vol. 28, 1990, pp. 553–560.

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

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

[13] 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.

[14] Zhou, Z. H.—Geng, X.: Projection Functions for Eye Detection. Pattern Recognition, Vol. 37, 2004, pp. 1049–1056.



Ghassan J. Mohammed received the M. Sc. degree in computer science from Mosul University, Iraq, in 2000. He held teaching positions at the College of Computer and Mathematical Sciences Department of Mosul University since 2001. He is currently pursuing the Ph. D. degree in computer science at Harbin Institute of Technology, Harbin, China. His research interests include image processing, pattern recognition and biometrics.

Bing-Rong Hong received his Ph. D. degree in computer science from Waseda University, Japan, in 1983. He is currently the Director of Multi-agent and Robotics Research Center at Harbin Institute of Technology, Harbin, China, where he has been a Professor since 1986. Also, he is presently the Director of Robot Soccer Technology Society of CAAI and Vice President of Federation of International Robot-soccer Association (FIRA), as well as the President of FIRA China Branch. His research fields include distributed AI, multi-agent robot systems, intelligent robotics, swarm Robot, virtual reality.

Ann A. Jarjes received the B. Sc. and M. Sc. degrees in computer science from Mosul University, Iraq, in 2001 and 2003, respectively. Since 2003 she has been a lecturer at the Department of Computer Science, Mosul University. She is currently pursuing the Ph. D. degree in computer science at Harbin Institute of Technology, Harbin, China. Her research interests include biometrics, image processing and AI.