

Iris Localization with an Improved Least Squares Fitting Algorithm

Tao Wei^{1,a}, Shuxiang Song^{1,b} and Haiying Xia^{1,c}

College of Electronic Engineering, Guangxi Normal University, Guilin, 541004, China

^ataotaotao110@163.com, ^bsongshuxiang@mailbox.gxnu.edu.cn, ^cxhyhust@gmail.com

Keywords: Iris Recognition, Iris Localization, Least Squares Fitting, Circular Arc Segment

Abstract. According to the disadvantages that calculation speed of traditional iris localization algorithm is slow and takes up much space. In this paper, we propose a new iris localization algorithm with an improved least squares fitting. Making use of the characteristics that gray value of iris image is difference from peripheral image around the iris, it uses Canny edge detector to detect iris image boundary and looks for a proper threshold to binarize image, and then uses least squares fitting to locate inner boundary and makes up circular arc segment of outer boundary. Finally, the center coordinate and radius of iris boundary can be easy to compute. The experimental results basing on over 150 images from CASIA iris image database have shown that accuracy of our localization method can be achieved as high as 99% and is over 150ms faster than the references mentioned in the paper. Our algorithm also reduces calculation complexity and saves memory space.

Introduction

With the rapid development of society, the importance of the identification has become increasingly prominent. In recent years, the effective identification and authentication of the identity requires more reliable, more accurate and more convenient in some important areas, such as in safeguarding state security, social security, network security, financial information security and so on. Just relying on traditional identification technique^[1] (just as passport, credit card) can't already satisfy the demands of authentication and identification accuracy in information era. Traditional identification techniques have lagged behind and the recognition means are too single. They are very difficult to distinguish the forgers and the real owners. Driven by the strong need, many new biometric identification technology emerge^[2], such as fingerprint recognition, face recognition, palm print recognition, voice recognition, iris recognition and so on. And iris recognition is one of the most potential market and business value in biological identification methods. It is a kind recognition method with non-contact, good anti- counterfeiting performance, high identification and exact localization.

Iris localization is that iris circle is extracted from the eye image by certain algorithms. It is very important step in the whole iris recognition process. Whether iris localization is accurate or not, directly affects the extraction of the iris texture feature and concerns the final result of iris recognition. At present, there are two main iris localization algorithms in iris recognition area: (1) Wildes proposed Hough transform^[3]. This method uses the edge points to vote for the boundary parameters. Because of heavy time-consuming with Hough transform, it is difficult to achieve real-time performance. (2) John Daugman proposed calculus circle detection algorithm^[4-5]. This method uses a circle detector to locate the iris boundary and searches the gradient and circular integral along the radial direction. After several iterative calculation, it would need more calculation. Reference [6] uses rectangular window to find iris boundary. Algorithm uses minimum average gray values to locate pupil. From the experimental result we find that this method is easy to make mistake when the light and shade of the image is great difference. In this paper, we propose a new iris localization algorithm. For inner edge location, it uses least square method (LSM) fitting circul-

*Corresponding author: songshuxiang. Tel.:+867735826559;fax:+867735826559. E-mail address: songshuxiang@mailbox.gxnu.edu.cn.

ar parameters, calculates center of the circle and the radius. For outer edge location, it uses point of intersection of different normal lines on the arc segment to draw up center of the circle, establishes polynomial equations. Then it uses the least squares fitting both left and right arc lines respectively, calculates the radius outer boundary and the center of the circle. Our algorithm is much simple and consumes less time.

Iris Image Pre-Processing

During collecting eye images, it is impossible to fix the iris in the same position. The image is much bigger than the eye when getting them through the camera. An eye image not only contains iris region but also includes other parts of the eye, such as eyelid, eyelashes, sclera and pupil. When getting an image, some features that contrast, light and shade degree are easily influenced in different condition. These differences will directly affect localization effect. So it is necessary to add a weight value for the iris images boundary before locate and perform binary image processing.

Iris Boundary Extraction

Iris boundary is similar to a circle. In order to effectively detect the iris boundary, we use different boundary direction operators to extract the boundary information of different direction, and use directional gradient operator to extract different direction gradient. We set four directional boundary gradient operators. Operators are defined as follows:

$$\begin{array}{ll} \begin{bmatrix} -1 & 0^* & 1 \end{bmatrix} & \text{(a) Dorizontal direction.} \\ \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0^* & 0 \\ 0 & 0 & 1 \end{bmatrix} & \text{(c) } 135^\circ \text{ Direction.} \\ \begin{bmatrix} -1 & 0^* & 1 \end{bmatrix}^T & \text{(b) Vertical direction.} \\ \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0^* & 0 \\ 1 & 0 & 0 \end{bmatrix} & \text{(d) } 45^\circ \text{ Direction.} \end{array} \quad (1)$$

Assuming iris image is $I(x,y)$, then four directions boundary of the images are described as:

$$\begin{cases} H(x,y) = I(x+1,y) - I(x-1,y) & \text{(a) Horizontal direction edge} \\ V(x,y) = I(x,y+1) - I(x,y-1) & \text{(b) Vertical direction edge} \\ D_1(x,y) = I(x+1,y+1) - I(x-1,y-1) & \text{(c) } 135^\circ \text{ direction edge} \\ D_2(x,y) = I(x-1,y+1) - I(x+1,y-1) & \text{(d) } 45^\circ \text{ direction edge} \end{cases} \quad (2)$$

Getting gradient image by weighing four directions gradient, as is shown in Formula (3):

$$\text{Grad}(x,y) = \sqrt{\alpha^2 \left(H(x,y) + \frac{D_1(x,y) + D_2(x,y)}{2} \right)^2 + \beta^2 \left(V(x,y) + \frac{D_1(x,y) - D_2(x,y)}{2} \right)^2} \quad (3)$$

In this formula, α, β are weighting coefficient, $\alpha, \beta \in (0,1)$, $H(x,y)$ is horizontal boundary, $V(x,y)$ is vertical boundary. $\alpha=1$ and $\beta=0$ are for the gradient of horizontal direction; $\alpha=0$ and $\beta=1$ are for the gradient of vertical direction. The gradient magnitude has the feature of same type. It is an effective way to detect different boundaries.

Boundary Gradient Binaryzation

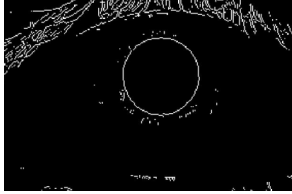
Image binarization is a simple way to segment image. This method is that setting one or more thresholds according to different grayscale of the image. It can be considered the similar pixels if they are within the same threshold level. In previous calculation, we get the magnitude and direction of iris boundary gradient through setting proper values for weighted coefficient of gradient formula. In order to solve the problem that iris image is interfered with speckle noise or is harder to determine with strong and weak points during extracting iris. After getting the gradient image, we choose Canny edge detector to further determine the boundary points. Calculation is divided into two steps:

- (1) Non-maximum suppression for gradient amplitude.
- (2) Detecting and connecting boundary algorithm with double thresholds.

Now, the binary boundary image equation is shown as:

$$B(x, y) = \begin{cases} 1 & \text{Grad}(x, y) > T_1 \\ & \text{or Grad}(x, y) > T_2 \text{ and Grad}(x', y') > T_1. \\ 0 & \text{others} \end{cases} \quad (4)$$

In the formula, $\text{Grad}(x, y)$ is boundary gradient, T_1, T_2 are the threshold what we set to detect strong borders and weak borders respectively. T_1, T_2 can effectively distinguish the noise boundary points and the iris boundary points by taking appropriate value. Using Canny algorithm to get the image binarization boundaries are shown in Fig.1.



(a) The iris inner boundary



(b) The iris outer boundary

Fig.1 Examples of iris boundary binarization image

Localization Inner Iris Boundary

An eye image mainly contains three parts, pupil, iris and sclera. The gray values of the three parts show an ever gradient increasing trend. The average grayscale of the pupil is minimum and more uniformity. The iris section contains rich texture. Gray value is bigger than the sclera. The sclera is in the outermost layer. Gray value is bigger than the iris. Iris location bases on these change rules to find boundary line and segment pupil, iris and sclera. From the previous image preprocessing, the iris image is implemented with edge detection and binarization. The iris inner boundary is not susceptible to interfere with of the eyelids and eyelashes. It forms a closed curve at the junction of the iris and the pupil. The image boundary shows clarity as shown in Fig.1 (a). Geometrically, iris inner boundary is similar to a circle. To locate the inner boundary we have to find the center and radius of the inner circle. We choose least-square method to locate iris inner boundary. Discrete points of the circle boundary are put in circle equation. In order to enable iris location more accurate we pick enough points on the circle to calculate. Algorithm is no longer to search on the whole image and perform multidimensional projection, but it just sets up equations in two-dimensional plane. Specific algorithm is described as follows:

There are known that the inner boundary coordinate values of the iris are (x_i, y_i) , $(i=1, 2, \dots, n)$, the circle equation as following:

$$(x - x_0)^2 + (y - y_0)^2 = r^2 \quad (5)$$

In the formula, (x_0, y_0) is the center coordinates, r is the radius, $(x, y) \in (x_i, y_i)$. Expanding and transposition formula (5) can get formula (6):

$$x^2 + y^2 = 2x_0x + 2y_0y + r^2 - x_0^2 - y_0^2 \quad (6)$$

In here, we set the parameters $b_1 = 2x_0$, $b_2 = 2y_0$, $b_3 = r^2 - x_0^2 - y_0^2$. The formula (6) can be transform as formula (7).

$$b_1x + b_2y + b_3 = x^2 + y^2 \quad (7)$$

With the help of calculation idea of the least squares method, the formula (7) is converted to a matrix form:

$$\begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} x_1^2 + y_1^2 \\ x_2^2 + y_2^2 \\ \vdots \\ x_n^2 + y_n^2 \end{bmatrix} \quad (8)$$

According to the related properties of non-homogeneous linear equations, the parameters $b_1, b_2,$

b_3 are calculated, and the center coordinate and radius can be fit by substituting these parameters values into formula(6). The result is shown as:

$$\begin{cases} x_0 = 0.5b_1 \\ y_0 = 0.5b_2 \\ r = \sqrt{b_3 + 0.25(b_1^2 + b_2^2)} \end{cases} \quad (9)$$

Thus we can locate iris inner boundary after to get the center coordinate and radius of the inner circle.

Localization Outer Iris Boundary

We know from Fig.1 (b) that the upper and lower portions of iris outer boundary are shaded by eyelids. It is unable to form a closed curve. If directly using least squares to locate outer boundary fitting circle like the way to locate inner boundary, it is likely to mistake eyelids boundary as iris boundary and cause misjudgment. From observing the left and the right sides of iris area, we can see they form two circular arc lines. They can be considered to be similar to a concentric arc. For such an incomplete circle, it is not suitable to calculate the circle radius and center coordinate by the way of fitting circle method. Locating the iris outer boundary, we use least squares fitting circular arc segment separately. We get three different points on the arc segments firstly and connect into two straight line segments, and then find the normal lines of the two line segments. The intersection of the two normal lines is the center of the arc. The implementation steps are given as follows:

As shown in Fig.2, taking three points on the right side arc curve as $D(x_d, x_d)$, $E(x_e, x_e)$ and $F(x_f, x_f)$, connecting the lines DF and EF. We set the line equations as $f(x) = kx + b$ and get the two pixels values of the points D,F into equation respectively and resolve equation $f(x)$. Making a normal line for the line DF with taking the middle point $M(x_m, y_m)$ of the points D,F. Normal equation is $f_1(x) = -\frac{1}{k}x + c$. Plugging the point M into $f_1(x)$ and calculating the parameter c.

In the same way, we can calculate the linear equation $f'(x) = k'x + d$, which go through the points E and F, and then calculate normal equation $f_2(x)$. Here making the two equations $f_1(x) = f_2(x)$, using the intersection between two normal lines, the center coordinate can be determined preliminarily by the right half circular arc curve DE. And then we use least squares fitting the right side of iris boundary section blocked by eyelid. Specific algorithm is described as follows:

We assume that arc curve DE polynomial is shown as: $h(x) = a_0 + a_1x + a_2x^2$, the parameters are a_0, a_1, a_2 . From the previous observing that the discrete pixels of arc segments are as (x_i, y_i) , $i=1,2,\dots,n$; In order to make circular arc curve much closer to iris boundary covered by eyelids. It has to make the sum of squared errors minimization. Setting $\varepsilon = h(x_i) - y_i$ and making formula (10) is minimum:

$$\sum_{i=1}^n \varepsilon_i^2 = \sum_{i=1}^n (h(x_i) - y_i)^2 = (Ax - y)^T (Ax - y) \quad (10)$$

in this formula,

$$A = \begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{bmatrix}, \quad X = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix}, \quad Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

For system $AX=Y$, equation may be compatible, or may be incompatible. In order to make sure the system is not an ill-condition system, left and right sides adding a multiplication actor A^T . System equation becomes $A^T AX = A^T Y$. It is referred to as normal equations of the system, the formula is described as:

$$\begin{bmatrix} n & \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i^3 \\ \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i^3 & \sum_{i=1}^n x_i^4 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n x_i^2 y_i \end{bmatrix} \quad (11)$$

The parameters (a_0, a_1, a_2) are calculated by resolving formula (11). Thereby the curve polynomial $h(x)$ is calculated too. When making $h(x)=0$, the roots of the equation $h(x)=0$ is calculated, thus it is easy to calculate the center coordinate and radius of the right arc. Using the same method, we fit the left trajectory curve BC and calculate the radius and the center of the circle corresponded by the left half arc BC. We find that the center of the left and the right side circular segments are essentially coincident through repeating experiments. This means that two pieces circular arc can be regarded as the same circle boundary. So the center coordinate (x_0, y_0) and the radius R of iris outer boundary can be determined.

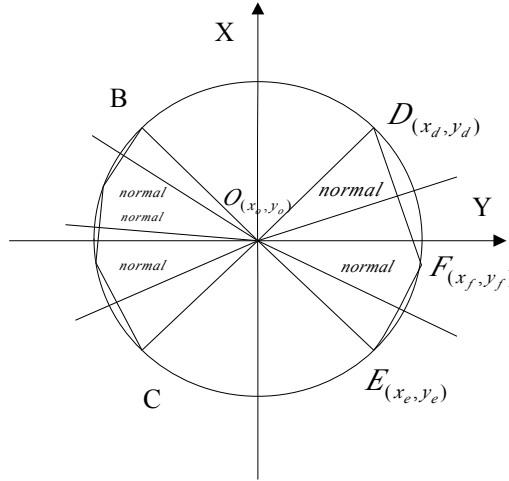


Fig.2 Arc fitting simulation image

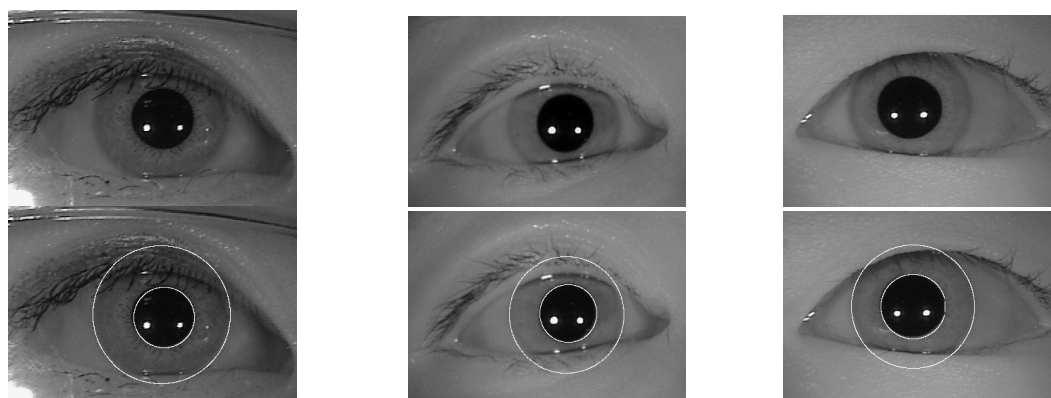
Experiment Result

In this experiment, we use iris images database provided by CASIA Iris Image Database Version 4.0, use MATLAB11b programming to realize the algorithm proposed in this paper, and carry out several experiments for the same image, also locate different images of the same eye. The experimental result shows that, when the brightness and clarity of the same eye picture is different, accuracy is basically the same except computing time is slightly different. In this paper, all the pictures are randomly adopted from CASIA 4.0. Experimental results are compared with other algorithms. The effect and speed of iris localization are presented in Table 1.

Table 1: Performance comparison of several algorithms

algorithm	Accuracy (%)	Inner Boundary Time(ms)	Outer Boundary Time(ms)
Hough Transfom	97.00	421	510
Daugman's Method	97.05	353	470
Reference[7]	97.55	235	316
Reference[8]	98.16	284	217
Proposed Method	99	135	210

From the table 1, we can see that the result of the algorithm that we propose is better than other traditional methods in the accuracy and speed. The results of iris localization are shown in Fig.3.



(a) S5457L09.jpg
Wearing Glasses

(b) S5116L06.jpg
The Left Eye

(c) S5132R01.jpg
The Right Eye

Fig.3 The results of iris image location

Conclusions

This paper has proposed an iris localization algorithm, this algorithm makes use of improved least square fitting to detect and locate inner and outer boundary of iris. It has been tested on CASIA 4.0 image database of 150 images. It has achieved the accuracy of 99%. The time-consuming is less than 400ms. What is more, it reduces computational complexity and saves memory space.

References

- [1] Jain, A.K., Ross, A., Prabhakar, S.: “ An introduction to biometric recognition ” , IEEE Trans.Circuits Syst. Video Technol., 2004, 14(1):4-20.
- [2] K.W. Bowyer, K. Hollingsworth, and P.J. Flynn, "Image understanding for iris biometrics: A survey", Computer Vision and Image Understanding, 2008,110(2): 281-307.
- [3] R.P.Wildes., “ Iris recognition:An emerging biometric technology,” Proceedings of the IEEE, 1997, 85(9):1348-1363.
- [4] J.G. Daugman, “High confidence visual recognition of persons by a test of statistical independence,”IEEE Transactions on Pattern Analysis andMachine Intelligence, 1993,15(11): 1148-1161.
- [5] John Daugman, “New Methods in Iris Recognition[J],” IEEE Trans.on syst.,Man,and Cybern. Part B, 2007, 37(5):1167-1170.
- [6] Yan Li Wen Li, Yide Ma, “Accurate Iris Localization Based on Region of Interest ” Biomedical Engineering and Biotechnology (iCBEB), 2012, pp.704-707.
- [7] Xiaobo Zhang,Zhenan Sun and Tieniu Tan,” Texture Removal for Adaptive level set based Iris Segmentation,” Proceedings of IEEE,pp, 1729-1732, 2010
- [8] Gao,Ting ; Xu,Zhong-Yu ; Yao,Qingan ,”An improved algorithm for Iris Location,” Mechatronics, Control and Electronic Engineering (CMCE), 2010,1:219-221.