Fast Algorithm for Iris Localization Using Daugman Circular Integro Differential Operator

Mahboubeh Shamsi, Puteh Bt Saad, Subariah Bt Ibrahim, Abdolreza Rasouli Kenari
Faculty of Computer Science & Information System, University Technology Malaysia, Johor, Malaysia
mahboubehshamsi@yahoo.com, drputeh@gmail.com, subariah@utm.my, rs.reza@gmail.com

***Abstract*—** An iris-recognition algorithm first has to identify the approximately concentric circular outer boundaries of the iris and the pupil in eye's image. We enhanced Daugman method to locate the iris. Iris is located by using a Average Square Shrinking approach. The algorithm is tested using iris images from CASIA database and MMU database. The circle contour sampling parameter has been investigated to find a tradeoff between speed and accuracy of algorithm. The experiments ensure the high speed of our algorithm related to other same methods. The spent time with high value of contour sample is 1.24s with 100% accuracy. Our approach is feasible for online authentication application that need more speed in detection.

***Keywords*—** Iris Segmentation, Daugman's Method, Average Square Shrinking, Circle Contour Sampling, Algorithm Speed ability.

I. INTRODUCTION

The iris is the most important and famous biometrics that useful for identification in the secure place. But the first and difficult step for this biometric is localization of this. However the iris segmentation process is challenging due to the presence of eye lashes that occlude the iris, the dilation of pupils due to different light illumination and several other uncontrolled factors.

Iris is a useful biometric for identifying individuals that fix during lifetime. Also the iris template contains an objective mathematical representation of the unique information. We use information for comparison between all of template that before stored in database. If this template can match to another subject in database, it's identified, if it no, it's unidentified.

There are four stages for identity the person with the iris. These are including: Preprocessing, Segmentation/iris localization, Filtration, Normalization, Feature encoding, Template matching.

The iris images consist of eyelid, eyelash, pupil, iris and sclera. Thus, the images must be preprocessed before feature extraction can be performed on those images. The size and contrast of the iris images also are not standardized since these are depended on lighting conditions and the distance between the eye and the iris camera. Hence the iris image should undergo a preprocessing step before further phases can be performed. The iris segmentation (localization) is proposed in this paper.

II. RELATED WORK

The first stage of iris recognition is to isolate the actual iris region. Two circles are used for separate between iris/sclera boundary and iris/pupil boundary. The eyelids and eyelashes normally occlude the upper and lower parts of the iris region.

The success of segmentation depends on the imaging quality of eye images [5]. Also, persons with darkly pigmented irises will present low contrast between the pupil and iris region if imaged under natural light, making segmentation more difficult [6]. An automatic segmentation algorithm based on the circular Hough transform is employed by Wildes et al. [15].

Daugman use a differential operator for locating the circular iris, pupil regions and the arcs of the upper and lower eyelids. The differential operator is defined as

$$\max_{(r,x_p,y_o)}\left|G_\delta(r)*\frac{\partial}{\partial_r}\oint_{r,x_o,y_0}\frac{I(x,y)}{2\pi r}d_s\right|$$

Where $I(x, y)$ is the gray level of image in pixel (x, y), $G_\delta(r)$ is Gaussian smoothing filter, s is the counter of circle represented by (x_0, y_0) as center and r as radius. The operator searches for the circular path where there is maximum change in pixel values, by varying the radius and centre x and y position of the circular contour. However, the algorithm can fail where there is noise in the eye image, such as from reflection.

Wildes et al. [16] also make use of the parabolic Hough transform to detect the eyelids, approximating the upper and lower eyelids with parabolic arcs, which are represented as;

$$\left(-(x-h_j)\sin\theta_j+(y-k_j)\cos\theta_j\right)^2 = a_j\left((x-h_j)\cos\theta_j+(y-k_j)\sin\theta_j\right)$$

Where a_j is the controls of the curvature, (h_j, k_j) is the peak of the parabola and θ_j is the angle of rotation relative to the x-axis. This method needs to choose threshold values for edge detection. Result may be fail for detecting the circles/arcs because this result maybe in critical edge points removes.

The method of Camus and Wildes [3] is similar to Duagman's method [4]. In this algorithm, they try to find the three circumference parameters (centre (x, y) and radius z) by maximizing the following function

$$C=\sum_{\theta=1}^{n}\left((n-1)\|g_{\theta,r}\|-\sum_{\phi=\theta+1}^{n}\|g_{\theta,r}-g_{\phi,r}\|-\frac{I_{\theta,r}}{n}\right)$$

Where n is the total number of directions and $I_{\theta,r}$ and g_{θ},r are, respectively, the image intensity and derivatives with respect to the radius in the polar coordinate system.

This algorithm works well when the regions between iris and pupil and iris and sclera are clearly separate. But if there is some noise or reflection in image, the result is not accurate.

Martin-Roche et al.'s method [10] is also the same as Daugman's operator. The average intensity difference of the five consecutive circumferences, as follows:

$$D = \sum_m \left(\sum_{k=1}^{5} \left(I_{n,m} - I_{n-k,m} \right) \right)$$

Where

$$I_{i,j} = I\left(x_0 + i\Delta_r \cos\left(j\Delta_\Theta\right), y_0 + I\Delta_r \sin\left(j\Delta_\Theta\right)\right) \cdot \Delta_r$$

and Δ_Θ are the increments of radius and angle, respectively, and $I(x, y)$ is the image intensity. Three circumference parameters (centre (x, y) and radius r) can be found, where the intensity difference between five successive circumferences is maximal. If the images do not have sufficient contrast between iris and sclera region, it can fail.

Tuceryan's method [14] use moments in small windows of the image. He segments the image by applying a clustering algorithm. The second-order regular geometric moments function for each pixel in the image is:

$$M_{pq} = \sum_{-w/2}^{w/2} \left(\sum_{-w/2}^{w/2} \left(I(m,n) x_m^p y_n^q \right) \right)$$

Where M_{pq} is the regular geometric moment of order pq, $I(m, n)$ is the pixel image intensity, x, y are the neighborhood window coordinates and W is the width. Tuceryan's algorithm some times does not have sufficient discriminate capacity.

III. Proposed Algorithm

Daugman operator [5] is based on the fact that the illumination difference between inside and outside of pixels in iris edge circle is maximum. It means that if you calculate the difference values of pixels' gray level in iris circle, this value is higher than any other circles in images. This fact turns to color of iris and color of sclera. Sclera is the white region outside of iris. See Figure 1.

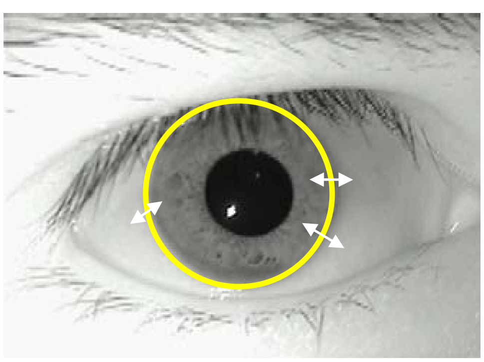

Figure 1. Difference between inside and outside of iris edge

It is impossible to calculate Daugman Operator for all feasible circle of an image. Therefore, we should restrict the space of potential circle. For example, in many research, they assume that the center of iris is near to center of image. Also they spot a range of radius base on size of image.

Let us explain the operator with an example, Figure 2.a shows a source of an eye image. This image has been selected from MMU iris database with 320*240 dimensions. We assume a square of 10*10 around the center of image as potential center of iris. We also assume a range of 55 to 65 pixels for potential radius. The potential centers and radius are shown in Figure 2.b.

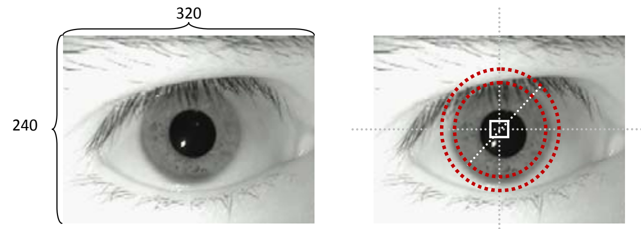

Figure 2. A-Source Eye Image, B-Potential Centers and Range of Radius

Now, for each (x, y) and all values of r, we should calculate the Daugman operator. The (x, y) and r which corresponded to higher value of Daugman operator will be a final center and radius. In our example, we have 100 potential centers and 10 values for radius. It means that we should calculate Daugman operator for 1000 potential circle. The algorithm is shown as follows:

```
For x = 155 to 165 do // (320 / 2) ± (10 / 2)
  For y = 115 to 125 do // (240 / 2) ± (10 / 2)
    For r = 55 to 65 do
      S = Daugman-Operator(x, y, r)
      If s > max then
        Register (x, y, r) as new values
```

Calculating Daugman operator, we need to compute the difference between inside and outside pixel values on contour of iris edge's circle. Figure 3 shows how we can compute Daugman operator for a circle with (x_c, y_c) as a center and r as a radius. Having the center point and the radius, a pixel on circle contour in angle α is computed as follows:

$$x = x_c + r \cdot \cos(\alpha), y = y_c + r \cdot \sin(\alpha)$$

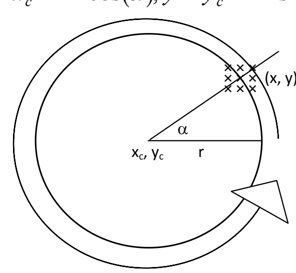

Figure 3. Computing Daugman Operator

We should calculate the difference of all pixels on circle contour and it is clearly impossible so we should adjust an upper limit *CS.* The *CS* assesses that how many points should be accessed on circle contour for computing Daugman's integral. It also corresponds to angle division around the circle contour. Higher value for *CS* clearly will reduce the estimate error of computing Daugman's integral.

Back to difference computing, we should calculate the difference value of each pixel in contour and then these values should be added together to compute Daugman's operator. We formulate the difference function as follows:

$$diff(x,y) = I(x+\Delta_\alpha, y+\Delta_\alpha) - I(x-\Delta_\alpha, y-\Delta_\alpha)$$

For coordinates (x, y) we have only integer values and we cannot calculate exactly difference values previously. We have only the values of top, bottom, left, right and diagonal pixels which are shown in Figure 3. Two points of diagonal pixels are in main diagonal and two of them on sideway. We also know that every point on one diagonal have 45 angle into themselves. We proposed the new formula to calculate four sensitive points that on the diagonal which is illustrated in bellow formula. By these explanations, we convert the Daugman operator as follows:

$$\max_{(x_c, y_c, r)} \sum_{j=1}^{CS} diff(x_j, y_j)$$
$$for\ all(x_c, y_c) \in potential\ centers\ \&\ r \in potential\ radius$$
$$x_{mj} = x_c + r \cdot \cos(\alpha_j),\ y_j = y_c + r \cdot \sin(\alpha_j),\ \alpha_j = 2\pi * {}^{j}/_{CS}$$
$$diff(x_j, y_j) = I_1 + I_2 + I_3 + I_4$$
$$I_1 = (I(x_j+1, y) - I(x_j-1, y)) \cdot \cos(\alpha_j)$$
$$I_2 = (I(x, y_j+1) - I(x, y_j-1)) \cdot \sin(\alpha_j)$$
$$I_3 = (I(x_j+1, y_j+1) - I(x_j-1, y_j-1)) \cdot \sin(45+\alpha_j)$$
$$I_4 = (I(x_j+1, y_j-1) - I(x_j-1, y_j+1)) \cdot \cos(45+\alpha_j)$$

To find the iris circle by Daugman integro differential operator, we need a potential centers, a range of radiuses and a number CS for sampling circle's contour. As we mentioned before, a possible center could be the center of image. But in many eye images, the center of iris doesn't fit to center of image. In next the section, we proposed a method for finding potential centers for Daugman operator.

A. *Average Square Shrinking (ASS) Approach*

As we mentioned before, computing Daugman's operator on all possible circle in eye's image is quite impossible. Therefore, we should restrict algorithm to a range of x, y as potential centers and a range of r as potential radius. Base on eye's structure, all eyes' pupil is black and base on this fact, the center of iris is black. Gaussian Blur or other smoothing method is applied for finding dark integrated pixels in image processing methods [11]. We also use this method for finding darkest place in eye image.

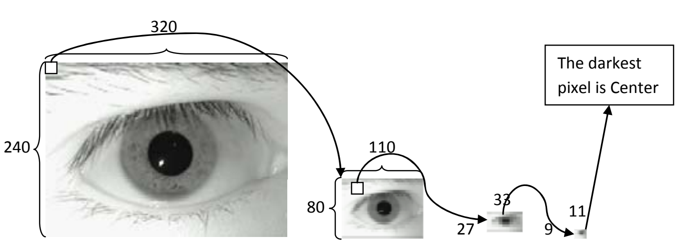

Figure 4. Shrinking Process

Our Average Square Shrinking (ASS) process is shown in figure 4. As you can see, each square in source image will be converted to one pixel in shrunken image. The size of square and the stages of shrinking are related to Shrinking (Smoothing) Factor S_f and Number of Shrinking Stages N respectively. These two parameters will be adjusted manually. The values of all pixels inside the square will be averaged in shrunken image. The darkest pixel (x_0, y_0) in last stage is the pupil center. The range of $[x_0 \pm S_f] \times [y_0 \pm S_f]$ will be used for potential centers of Daugman operator.

The range of radius, same as before, should be estimated manually. Having potential centers and estimated range of radius, it is now ready to apply Daugman operator for improving the iris center and radius. The higher value of Daugman operator will be corresponded to exact center and radius of iris. These steps will be continued iteratively on higher shrunken image to find final center and radius. The algorithm and flow chart are shown in figure 5.

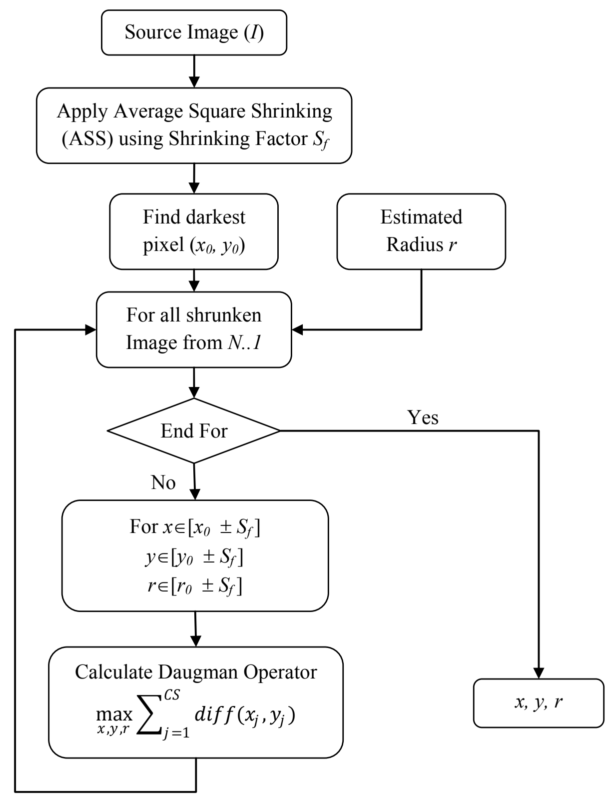

Figure 5. Proposed Algorithm

As summarized, we should adjust four parameters manually in our algorithm. These four parameters are:

1. Shrinking (Smoothing) factor S_f
2. Number of Stages N
3. Estimated radius r
4. Circle Contour Sampling CS

The estimated potential centers of Daugman method have been replaced by the first two parameters. With improvement, our potential centers are closer to real center than before. Daugman method also needs the last two parameters.

IV. Experimental Result

The proposed algorithm has been developed using Delphi programming language. It is tested on 2.4 GHz CPU with windows vista and 2 GB ram. Two famous iris databases have been selected for experiments. CASIA-IrisV3 [1] includes three subsets which are labeled as CASIA-IrisV3-Interval, CASIA-IrisV3-Lamp, CASIA-IrisV3-Twins. CASIA-IrisV3 contains a total of 22,051 iris images from more than 700 subjects. All iris images are 8 bit gray-level JPEG files, collected under near infrared illumination. MMU iris database [2] contributes a total number of 450 iris images. They come from Asia, Middle East, Africa and Europe. Each of them contributes 5 iris images for each eye.

As we mentioned before for calculating Daugman operator, we need to compute grayscale difference values of inside and outside pixels around all point of circle's contour. This computation is quite impossible. Therefore, we select some sample point of circle's contour. We named this factor by *CS* in our algorithm. The high value for *CS* will increase accuracy of method. But it also increases the time of computation. On the other hand, selecting a low value for *CS* will increase the fault segmentation, though the time will decrease. Figure 6 shows our experiments for different value of *CS*. First we select *CS* as 128 and without changing other parameters, we change *CS* by half. We segment images by *CS* as 64, 32, 16 and 8 respectively.

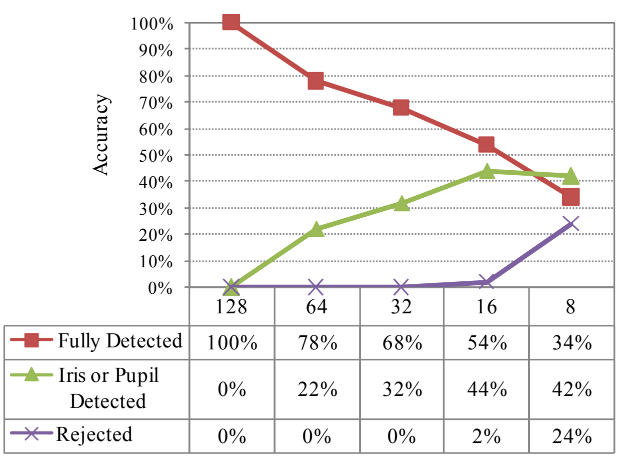

Figure 6. Accuracy percentage according to invariance of *CS*

As you can see, the error rate and false detection are significantly increased. So with value of *CS* below than 16, some images cannot be detected. Whereas with value of *CS* more than 16, the errors has been occurred only on iris or pupil, not both of them.

The theoretic base and our experiments show that by increasing *CS*, the accuracy will be raised, but it affect strongly on time. A longtime detection is not desirable for real life application. Figure 7 show the time of execution with above values of *CS*. Low variation of execution time ensures the practicability of algorithm. The result shows with higher value of *CS*, the average time of algorithm execution is only 1.24 s.

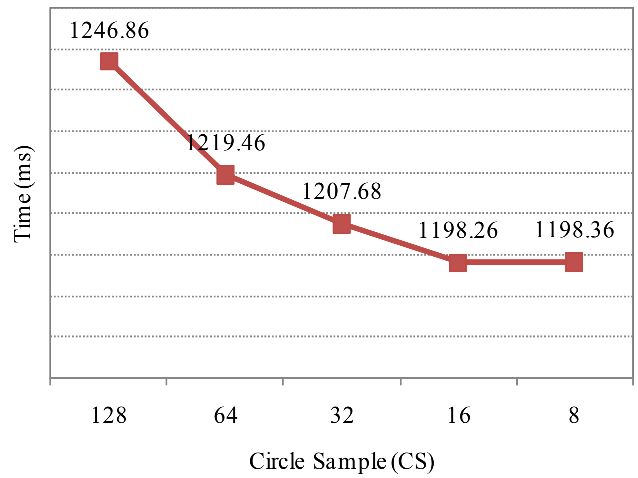

Figure 7. The time of execution with various values of *CS*

On the other hand, base on our claim, we compare our execution time with most famous segmentation algorithms. These algorithms are described before, Daugman's method [4] as base of many segmentation algorithm, specially our method, had registered 2.73s as its best average time. The best average time of Wilds [16] is 1.95s. They used Hough transform as base as their localization. Camus and Wildes [3], Martin-Roche et al. [10] and Tuceryan [14] also record 3.12s, 2.91s and 4.81s for their execution time, respectively. Whereas our average time of segmentation process in worst state, with higher value of *CS* is 1.24s. It shows that, our algorithm is faster than mentioned method with higher accuracy. The execution times of all methods [7, 12] are shown in Figure 8 to compare with each other.

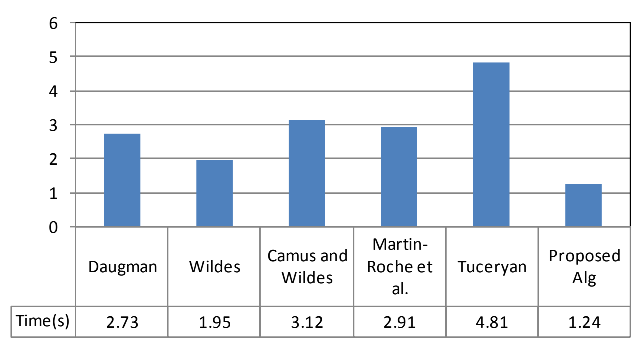

Figure 8. The execution times of all methods

The experiments show that selecting high-value for *CS* increases the accuracy and have little effect on the execution time, so we can ignore this change.

V. Conclusion

In this paper, we localize and select the Daugman Integro Differential Operator as base of algorithm. A discreet operator will be represented for Daugman operator to be implemented. For enhancing Daugman method, the first problem was locating the potential centers. We proposed a Average Square Shrinking (ASS) approach for offering the primary potential centers.

In our approach we focus on circle sampling *CS* in proposed method. The algorithm has been developed and experiments have been tested for finding a good tradeoff between accuracy and time. High value of *CS* will increase time and low value of *CS* will decrease the accuracy. Approach results in comparing with famous methods of segmentation shows that our method executes with desirable speed. It ensures that proposed method is suitable for less time application.

VI. Future Work

Some suggestion will be introduced for improving proposed algorithm and decreasing manual interference of user. These suggestions are as follows:

- Deleting the Number of Stages Parameter *N*
  - This can be possible by employing a threshold for size of last shrunken image. We continue the shrinking until the size of image will be less than threshold, for example 10 pixel.
  - This improve also could help us for deleting the range of radius parameter, because by investigating the eye structures, in an eye image with width 10 pixel, the radius can be estimated 2 pixel. Iris size related to eye size is like 4 to 10.
- Improving the difference function
  - Applying a good high contrast method for increasing the contrast of image around the circle contour to find circle more accurately
  - Applying another edge detection method same as Hough transform [8, 9, 13, 15].

Acknowledgment

This project was supported by Ministry of Science, Technology and Innovation (MOSTI) through the E-Science Fund grant (Vote79307).

References

[1]. *CASIA iris image database*, in http://www.sinobiometrics.com, Chinese Academy of Sciences Institute of Automation.

[2]. *MMU iris image database,* , in *http://pesona.mmu.edu.my/~ccteo*, Multimedia University.

[3]. Camus, T.A. and R. Wildes, *Reliable and fast eye finding in closeup images.* IEEE 16th Int. Conf. on Pattern Recognition, Quebec, Canada, 2004: p. 389-394.

[4]. Daugman, J.G., *High Confidence Visual Recognition of Persons by a Test of Statistical Independence.* IEEE Trans. Pattern Anal. Mach. Intell., 1993. **15**(11): p. 1148-1161.

[5]. Daugman, J.G., *Biometric personal identification system based on iris analysis.* United States Patent, 1994. **5**: p. 560.

[6]. Daugman, J.G., *How iris recognition works.* Proceedings of 2002 International Conference on Image Processing, 2002.

[7]. Jen-Chun, L., P.S. Huang, C. Chien-Ping, and T. Te-Ming. *Novel and Fast Approach for Iris Location*. in *Intelligent Information Hiding and Multimedia Signal Processing, 2007. IIHMSP 2007. Third International Conference on*. 2007.

[8]. Kong, W. and D. Zhang, *Accurate iris segmentation based on novel reflection and eyelash detection model.* Proceedings of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing, Hong Kong, 2001.

[9]. Ma, L., Y. Wang, and T. Tan, *Iris recognition using circular symmetric filters.* National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, 2002.

[10]. Martin-Roche, D., C. Sanchez-Avila, and R. Sanchez-Reillo, *Iris recognition for biometric identification using dyadic wavelet transform zero-crossing.* IEEE Aerosp. Electron. Syst. Mag, 2002: p. 3–6.

[11]. Mazur, J., *Fast Algorithm for Iris Detection.* Advances in Biometrics, 2007: p. 858-867.

[12]. Proenca, H. and L.A. Alexandre, *Iris segmentation methodology for non-cooperative recognition.* Vision, Image and Signal Processing, IEE Proceedings, 2006. **153**(2): p. 199-205.

[13]. Tisse, C., L. Martin, L. Torres, and M. Robert, *Person identification technique using human iris recognition.* International Conference on Vision Interface, Canada, 2002.

[14]. Tuceryan, M., *Moment based texture segmentation.* Pattern Recognit. Lett., 1994. **15**: p. 659-668.

[15]. Wildes, R., *Iris recognition: an emerging biometric technology.* Proceedings of the IEEE, 1997. **85**(9).

[16]. Wildes, R., J. Asmuth, G. Green, and S. Hsu, *A system for automated iris recognition.* Proceedings IEEE Workshop on Applications of Computer Vision, 2004.