



ISSN: 0067-2904

Iris Identification Based on the Fusion of Multiple Methods

Asaad Noori Hashim*, Roaa Razaq Al-Khalidy

Department of Computer Science, Faculty of Computer Science and Mathematics, University of Kufa, Najaf, Iraq

Received: 13/6/2020

Accepted: 24/7/2020

Abstract

Iris recognition occupies an important rank among the biometric types of approaches as a result of its accuracy and efficiency. The aim of this paper is to suggest a developed system for iris identification based on the fusion of scale invariant feature transforms (SIFT) along with local binary patterns of features extraction. Several steps have been applied. Firstly, any image type was converted to grayscale. Secondly, localization of the iris was achieved using circular Hough transform. Thirdly, the normalization to convert the polar value to Cartesian using Daugman's rubber sheet models, followed by histogram equalization to enhance the iris region. Finally, the features were extracted by utilizing the scale invariant feature transformation and local binary pattern. Some sigma and threshold values were used for feature extraction, which achieved the highest rate of recognition. The programming was implemented by using MATLAB 2013. The matching was performed by applying the city block distance. The iris recognition system was built with the use of iris images for 30 individuals in the CASIA v4. 0 database. Every individual has 20 captures for left and right, with a total of 600 pictures. The main findings showed that the values of recognition rates in the proposed system are 98.67% for left eyes and 96.66% for right eyes, among thirty subjects.

Keywords: Iris Identification, Scale Invariant Features Transform(SIFT), Local Binary Pattern (LBP), Difference of Gaussian (DoG).

تحديد القرنية بناءً على دمج طرق متعددة

اسعد نوري هاشم* , رؤى رزاق الخالدي

قسم الحاسبات، كلية علوم الحاسوب والرياضيات، جامعة الكوفة، النجف، العراق

الخلاصة

احتل التعرف على قرنية العين مرتبة مهمة في أنواع البيومترية نتيجة الدقة والكفاءة التي يتمتع بها. الهدف من هذه الورقة هو اقتراح نظام مطور لتحديد القرنية مبني على تحويل السمات الثابتة النطاق والنمط الثنائي المحلي لاستخراج الميزات. تم تطبيق عدة خطوات. أولاً، تحويل أي نوع صورة إلى نوع رمادي. ثانياً، توطين القرنية باستخدام تحويل هوغ الدائري. ثالثاً، كانت الخطوة هي التطبيع لتحويل القيمة القطبية إلى الديكارتيية باستخدام نماذج ورقة المطاط داوجمان، بعد ذلك باستخدام معادلة الرسم البياني لتعزيز منطقة القرنية. أخيراً، تستخدم استخلاصات الميزات تحويل السمات الثابتة والنمط الثنائي المحلي. تم استخدام بعض قيم سيجما وعتبة لميزة الاستخراج، والتي حققت أعلى معدل التعرف. تم تنفيذ البرمجة بواسطة (MATLAB) (2013). أداء المطابقة حسب مسافة كتلة المدينة، وبناء نظام التعرف على قرنية العين باستخدام صور القرنية لـ 30 فرداً في CASIA v4. 0 قاعدة بيانات، كل فرد لديه 20 لقطة لليسر واليمين، وإجمالي 600

*Email: asaad.alshareefi@uokufa.edu.iq

صورة ، والنتيجة الرئيسية لمعدلات التعرف في النظام المقترح هي 98.67% للعين اليسرى و 96.66% للعين اليمنى لثلاثين شخصًا.

1. INTRODUCTION

Iris recognition has gained more importance as a result of several applications, with high case consistency and highly impeccable recognition rates. Iris recognition is used in high-security environments, including boundary control at airports and harbors as well as laboratory contact control. Several papers have been published, with weaknesses and strengths. Rashad *Et al.* (2011) suggested an iris recognition and classification algorithm, using a framework built on local binary pattern and histogram properties. The algorithm was applied as a statistical method for features extraction, where a combined learning vector quantization classifier was used as a neural network classification tool. The identification rate for various iris datasets was 99.87 % as compared to different methods [1]. Gongping *et al.* (2012) proposed a new approach to recognize iris by invariant transformation feature scale. The experimental results showed that the invariant scale feature transformation algorithm, with enhancement and normalization, can especially improve recognition accuracy [2]. Harinder and Sunil (2016) suggested an approach for iris feature extraction using invariant scale feature transform, which is invariant in scale, somewhat invariant in rotation, and shows robustness in affinity distortion. The advantages of the proposed method are its accuracy and simplicity [3]. Rathgeb *et al.* (2018) proposed a method for iris recognition by scale invariant transform feature, representing a general purpose image descriptor, discriminative orientation based feature selection, and magnitude possibility distribution function. The weight assignment for the iris texture sub-regions showed an improved performance with equal error rates of 0.88%, and 0.9% for CASIAv3 and MMU, respectively [4]. Divya and Urmila (2016) used Daugman's method to determine the pupil and the iris borders. This work focuses more on an effective and accurate method for iris segmentation [5]. Humayan *et al.* (2019) proposed a technique that includes discrete wavelet transformation- principal component analysis to extract the optimization features for the iris. Our experimental estimate validates the successful implementation of the proposed method [6]. In this paper, we suggest a new approach built on SIFT as well as LBP, where SIFT produces a matrix of coefficients that has a variant size related to the size of the enrolled images. The proposed method induces an extensive acceleration, causing lower operation times which are equivalent to those of the conventional scheme. In addition, the proposed method proves to maintain an effective application of the biometric blend, achieving important performance increases in a difficult multi-algorithm blending with conventional scheme. Finally, SIFT proved to be more important as a result of its invariance to illumination, scale, noise, rotation... etc.

2. MATERIALS AND METHODS

Several approaches were proposed to handle iris recognition; some of them deal with normalization problems, while others deeply treat the recognition operation, as described below.

2.1. Iris Localization and Separation

Localization of iris is an important stage in the recognition scheme of human iris, with the aim of creating an exact allocation of iris borders. For the purpose of accuracy, this step will govern all the subsequent phases. Iris of human be located an annular part among Human iris is located at the annular part between the pupil (inner circle) and the sclera (outer circle), with both the outer and inner borders of the iris representing circles [7]. Finding the centers of the pupil and iris is an important stage in iris recognition.

2.2. Normalization of Iris

This process denotes the preparation of an iris image for the extraction of features. Illumination takes a direct effect on the size of the pupils and induces a non-linear pattern of iris variations. To offset these variations, a proper technique of normalization is required. The most widely employed model is the Daugman's rubber sheet, which transforms the circular region of the iris to a rectangular tablet of a fixed size, using equation (1). This model transforms all pixels in the circular iris into a corresponding position of the polar axes (r, θ), where (r) is the radial space and (θ) is the angle of rotation at the equivalent radii [8], as shown in Figure-1.

$$I(x(r, \theta), y(r, \theta)) \rightarrow I(r, \theta) \quad (1)$$

such that

$$x(r, \theta) \rightarrow (1 - r)Xp(\theta) + rx(\theta) \quad (2)$$

$$y(r, \theta) \rightarrow (1 - r)Yp(\theta) + ry(\theta) \quad (3)$$

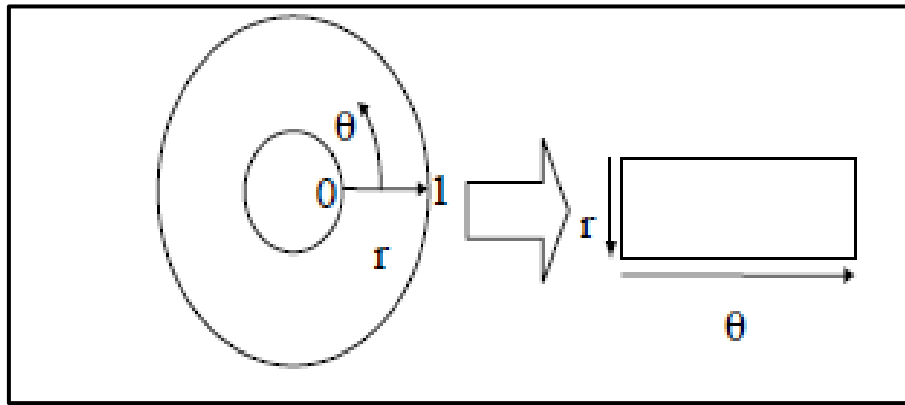


Figure 1-Polar to Cartesian mapping [8].

where $I(x, y)$ corresponds to the iris region. (x, y) and (r, θ) correspond respectively, transform cartesian to polar coordinate, where (θ) has a range of 0 to 2π and (r) has a range of R_p to R_i . The pupil boundary points are defined as the linear combination of $x(r, \theta)$ and $y(r, \theta)$, while (x_p, y_p) is the pupil coordinate of iris.

2.3 Feature Extraction

Extraction of the features is the key assignment in every confirmation method. Choosing an effective feature extractor component is the most critical factor in achieving high authentication rates in the iris identification system. Each iris picture has a special feature that is different from the others. Consequently, it is possible to solve a number of problems in pattern recognition by choosing a better function space. The system is implemented by using two different sets of feature extractors, which are proposed to create the features vector [9]. The features extractors utilized in this paper include the SIFT and the LBP.

2.3.1 Scale Invariant Feature Transforms

SIFT was suggested by David G. Lowe, the University of British Columbia, in 2004, using distinctive image features of key points. The method extracts key points and computes their descriptors. In the past ten years, SIFT has been widely used in many applications; for example, objects detection, objects classification, stereo correspondence, and motion tracking. The feature extraction is invariant to image scales and rotation, and was shown to provide robust matching across an extensive range of fine distortion, alteration in 3D viewpoint and noise, and change in illumination [10,11]. SIFT algorithm with its four stages is shown below.

2.3.1.1 Detection of Scale Space Extreme

The main phase of the SIFT algorithm examines the complete scales and image locations. The scale space of an image is the convolution variable scale with a Gaussian kernel function $G(x, y, \sigma)$ and the input image $I(x, y)$, where (σ) is scale space factor and (x, y) are space coordinates. The scale-space at different scales is shown in Figure-2 [12].

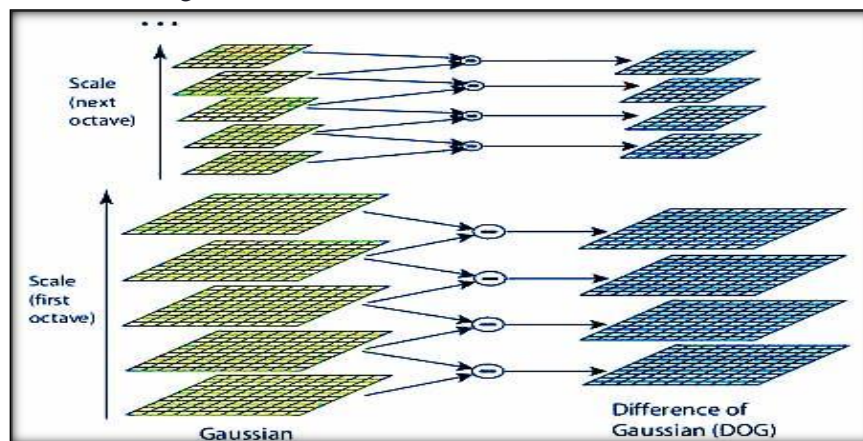


Figure 2-Gaussian-smoothed images and difference of Gaussian (DoG) [12].

The implementation was performed by using difference of Gaussian (DoG) to find potential interest points that are invariant to scales and orientation. DoG is calculated as an Eq.4 and Eq.5 [13]:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{4}$$

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \tag{5}$$

where,

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{6}$$

Algorithm of Scale-Space Extrema Detection (Building a pyramid and DoG)	
Initialization:	sigma // the amount of blur level in the image. Octaves // number of octaves required for DoG. Levels // number of levels in the octave. Row, Column // new size for the iris image .
Input:	img() // the digital image of size (M*N) pixels.
Output:	Temp-D // the difference of Gaussian (DoG) pyramid.
<p>Step1: read image and convert type image to gray. Step2: convert the size of image (m, n) to equal size as Row, Column. Step3: build the pyramid by down sampling. D{i} \ \ structure of three cells; the first cell content array size is (Row *Column) Pixels, second cell array size is (1/2Row * 1/2Column) pixels, and third cell array size is (1/4 Row *1/4 Column) pixels. The result is three images of different size. Step4: Create the DoG for each image in pyramid to detect the blob point: For each octave For each level in octave 1- Compute scale per octave by equation: Scale= sigma*(sqrt (2)).^(1/level). 2- Design Gaussian filter with scale factor. 3- L1=img() \ \ original image. 4- L2 (x, y, sigma) = G (x, y, sigma) * I(x, y) \ \ L2 convolve image with Gaussian filter. 5- Temp-D = L2 - L1 \ \ calculating the difference between two images. 6- Return Temp-D.</p> <p>End Algorithm</p>	

2.3.1.2 Accurate Localization Extreme Points

To locate the extreme points of scale space, every sampling point is compared with all neighboring points to realize if the point in its image domain and together scale is larger or smaller than the other points. The sampling point in the center of its neighbors relating a pixel to its 26 neighbors in 3x3 regions at present and adjacent scales, as shown in Figure-3. Key points are designated and founded with measurements of their stability. Thus, key points of low contrast and edge response instability are eliminated. Eigenvalues of the Hessian matrix (H) are computed to eliminate the edge responses, where H is given as in Eq.7, h11 and h22 as in Eq.8, and h12 and h21 as in Eq.9 [14]:

$$H = \begin{bmatrix} h11 & h12 \\ h21 & h22 \end{bmatrix} \tag{7}$$

where

$$\left\{ \begin{array}{l} h11 = (W_{m+1;n} + W_{m-1;n} - 2W_{m;n}) \\ h22 = (W_{m;n+1} + W_{m;n-1} - 2W_{m;n}) \end{array} \right\} \tag{8}$$

$$\left\{ \begin{aligned} h_{12} = h_{21} = & \frac{(W_{m+1;n+1} - W_{m+1;n-1} - W_{m-1;n+1} + W_{m-1;n-1})}{4} \end{aligned} \right. \quad (9)$$

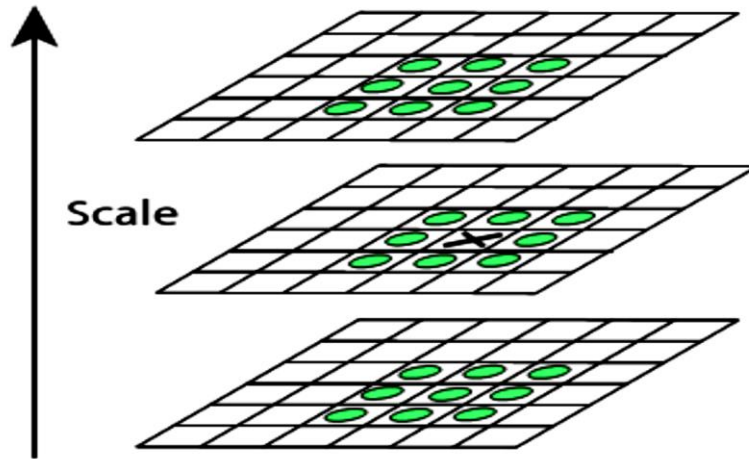


Figure 3- Local extreme detection in DoG scale space [12].

2.3.1.3 Orientation

All key point positions are giving one or more orientations, depending on the direction of the local image gradient. A gradient-oriented histogram (HOG) is created for each key point, which calculates gradient magnitude (m). The orientation histogram takes 36 bins cover and an orientation range of 360 degrees, and computes the orientation (θ). It then detects the highest peak in the histogram as well as some other local peaks within 80 % of the highest peak, as shown in Figure-4. The values of (m) and (θ) are computed by Eq.10 and Eq.11, respectively [15].

$$m(x, y) = \sqrt{[L(x + 1, y) - L(x - 1, y)]^2 + [L(x, y + 1) - L(x, y - 1)]^2} \quad (10)$$

$$\theta(x, y) = \tan^{-1} \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \quad (11)$$

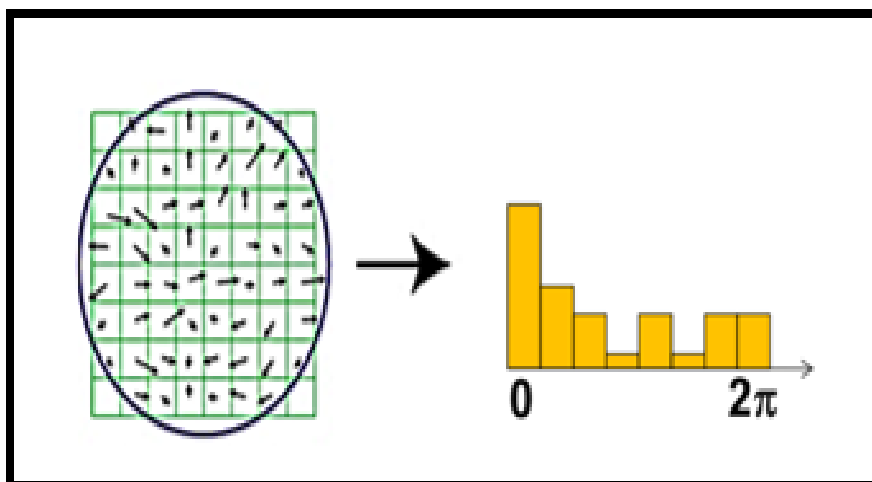


Figure 4- Image gradient histogram angle [12].

2.3.1.4 Key point descriptor

Once orientation has been designated, the features descriptor is calculated as in the histograms on (4 x 4) pixel neighborhood orientations, which are set as shown in Figure-5. The histograms of orientation are compared to the orientation of a key point. The histogram covers eight bins, and each descriptor contains an array around the key point of our histogram. This generates a SIFT feature descriptor of (4 x 4 x 8 = 128) elements, while the vector of the descriptor is invariant to rotation, scaling, and lighting [16].

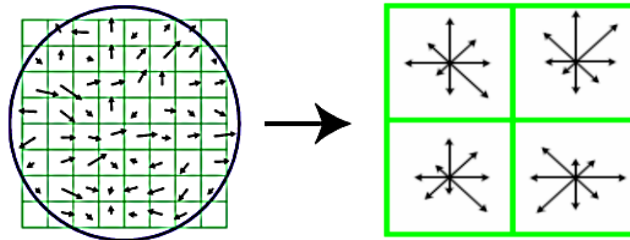


Figure 5- Key point descriptor [13].

2.3.2 Local Binary Patterns

Local binary pattern is used to extract local binary pattern histogram (LBPH) feature for the description of texture. Ojala introduced an LBP operator for texture classification, where he proposed using a two level form of the technique (Texture Spectrum). This offers a robust way to describe pure local binary patterns in a texture. There are only ($2^8 = 256$) possible texture units in the two-level version. In binary instance, the original (3 x 3) area, as shown in (Figure- 6), is thresholded via the value of the center pixel (each pixel as a threshold), i.e. its (3 x 3) region is transferred into an 8-bit binary code, as in the following condition :

$$(Ni, Nc) = 1 \text{ when } Ni \geq Nc, \text{ else } C(Ni, Nc) = 0$$

where (Nc) is the value of center pixel, (Ni) is the value of neighbor pixel, and (i) is location of neighbor pixel (1,2,3, ..., 8). The values of the eight pixels are summed to obtain the number of this texture element, as n Eq.12 [17].

$$LBPr_{row} = \sum C(Ni, Nc) \times 2^{i-1} \tag{12}$$

<i>N1</i>	<i>N2</i>	<i>N3</i>
<i>N8</i>	<i>N0</i>	<i>N4</i>
<i>N7</i>	<i>N6</i>	<i>N5</i>

Figure 6- (3 x 3) window of neighbor pixel.

2.4 The Proposed System

All the steps of the processing system are explained in the following diagram (Figure- 7).

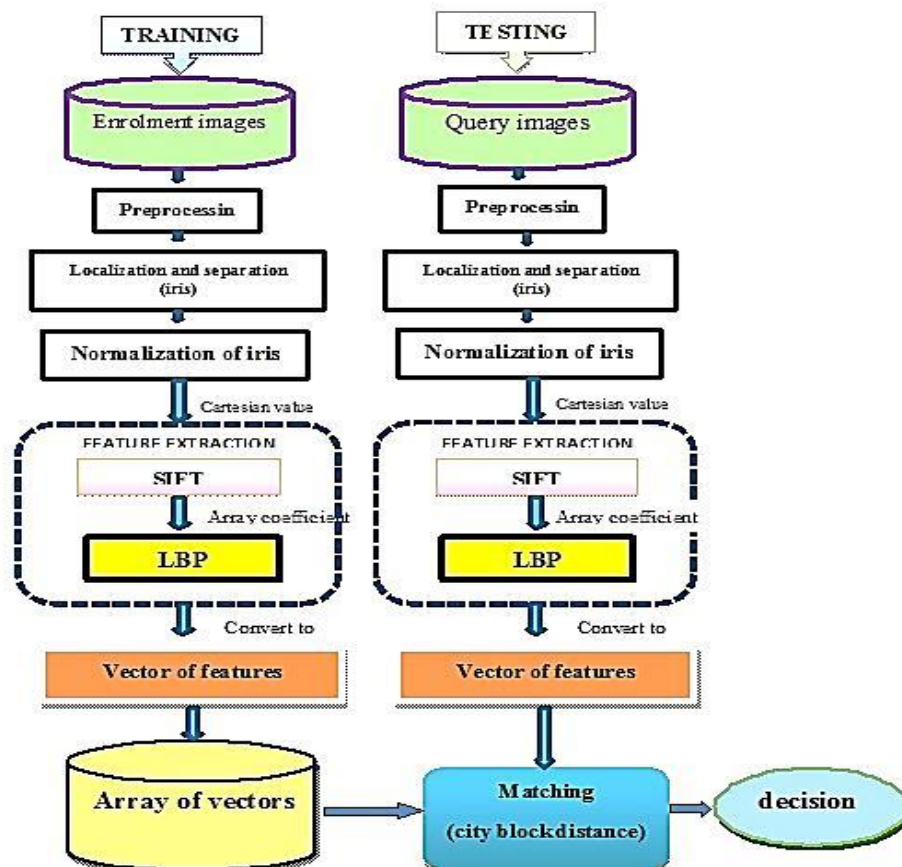


Figure 7- Diagram of the proposed system.

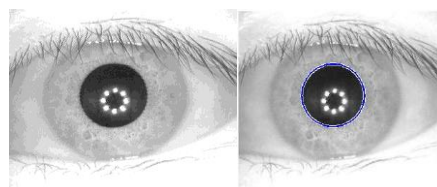
Steps of the processing system are:

1- Preprocessing: The type of the enrolled image was converted to gray level.

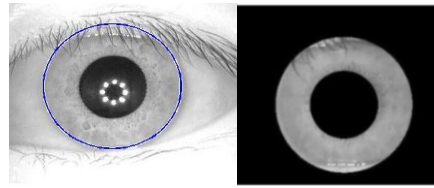
2- Localization and separation of IRIS: This step was used for the separation of iris from eye image. Circular hough transforms was utilized to find circles in pictures. This method is employed in the presence of noise, occlusion, and varying illumination, due to its robustness. Circular hough transforms function was employed to find coordinates of the center of pupil and iris, the radius of pupil (R_p), and the radius of iris (R_i). The next step is the usage of coordinates of the pupil (x_p , y_p) and coordinates of the iris (x_i , y_i) to cut the iris region of the entire eye. The radius (R) for each point in the eye image was determined using Eq.13 [7].

$$(R) = \sqrt{(y_i - y_p)^2 + (x_i - x_p)^2} \quad (13)$$

The value of (R) is compared with those of R_p and R_i , if it is between R_p and R_i , then it will be treated as an iris point, otherwise it is a non-iris point which is marked by zero value. Figure-8 indicates the method for the separation of the iris boundary.



(a) Original Image. (b) Pupil localization



(c) Iris localizations (d) Iris Separation

Figure 8- Localization and separation, (a) The Original Image, (b) pupil localization (c) Iris localizations, (d) The Iris Separation.

3- The Normalization Process: this process was accomplished by using Eq.14 and Eq. 15[8].

$$I(x, y) \rightarrow I(r, \theta) \rightarrow I(x_f, y_f) \tag{14}$$

such that,

$$x_f = \frac{NW \times \theta}{2\pi}, \quad y_f = \frac{NH \times (r - R_p)}{R_i - R_p} \tag{15}$$

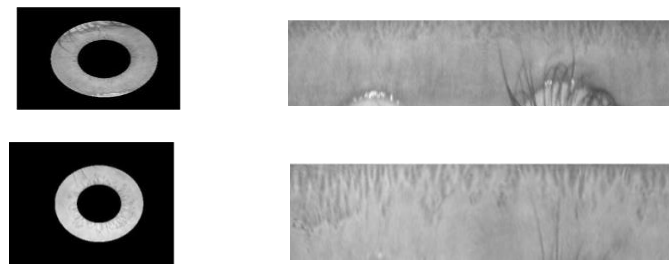
The polar coordinate (r, θ) , is determined by the pixel coordinates in the original iris image, as shown in Eq.16, Eq.17, and Eq.18 [8].

$$\theta = -\tan^{-1}\left(\frac{\Delta y}{\Delta x}\right), \quad r = \sqrt{(\Delta x)^2 + (\Delta y)^2} \tag{16}$$

$$x = x_p + r * \sin(\theta); \quad y = y_p - r * \cos(\theta) \tag{17}$$

$$\Delta x = x - x_p; \quad \Delta y = y - y_p \tag{18}$$

Figure-9 shows a sample image by applying the circular to rectangular normalization procedure.



(a) (b)

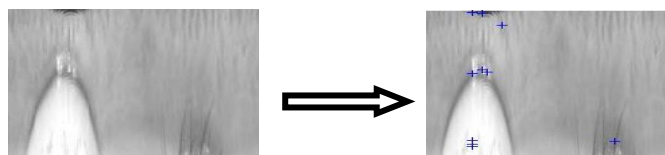
Figure 9- Normalization of the (a) Original image, (b) Normalized image.

AN example of one difficulty in iris identification is the occlusion that occurs because of the eyelids and eyelashes, as illustrated in Figure-10.



(a) Detected iris

(b) Normalized iris with a size of (361,720) pixels



(c) Resizing of the normalized

(d) Extraction of features.iris with the size of (200,200) pixels

Figure 10- Resizing of Iris Image and Features Extraction with Occlusion (Eyelids and Eyelashes)

This occlusion increases the difficulty and impacts efficiency extraction features represented by the blue color. It also causes error in the matching processes. Therefore, we proposed an approach of several iris regions to select a region of interest (ROI) for the iris area while evading the regions that could obstruct features extraction. This region must be with equal coordinates (row, column) to build a DoG pyramid. In this work, we used many areas of iris image which were imposed for experimentation, as shown in Figure-11. Regions that have no eyelids and eyelashes were selected. Also, this process decreased the time of feature extraction, when using sub-regions of iris (ROI) instead of the entire region of iris.

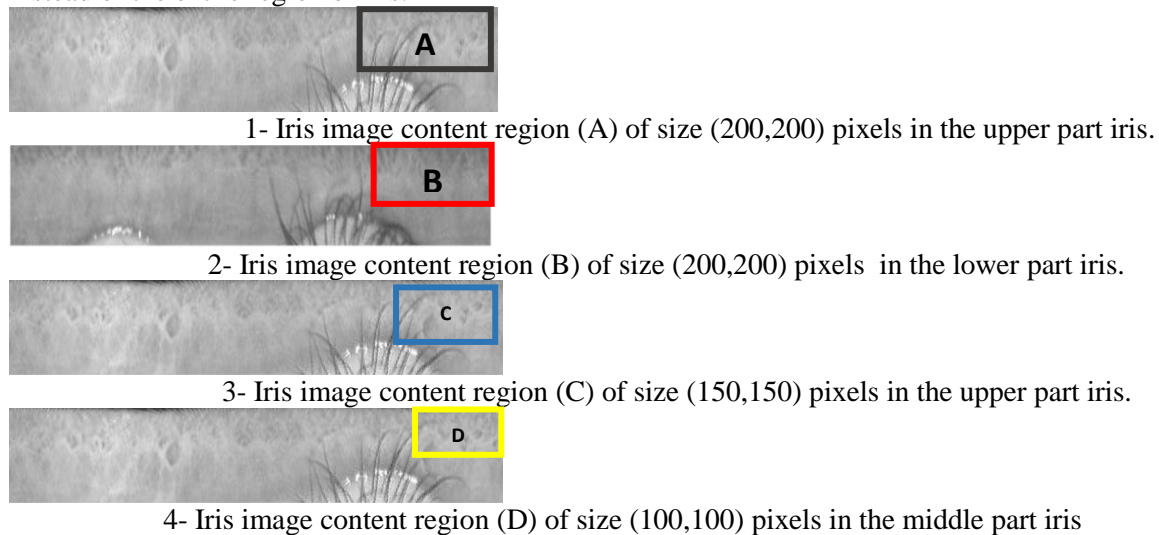


Figure 11- Regions of interest in the iris.

4- Feature extraction: The descriptor vector was formed for each key point. The local feature of the iris image around each key point was extracted by utilizing SIFTs. For each key point created, the descriptor size is $128 = (4 \times 4 \times 8)$ elements. After that, a coefficient array of features was obtained and converted to local binary pattern, while the results were stored in the feature vector. Finally, a row vector (Iris Code) was created and the feature templates were saved. To reduce image contrast and the effects of changing the lighting conditions, we used the local binary pattern, where each pixel's value varies according to the spatial relationship of the 8 pixels surrounding it [17], as illustrated in Figure-7.

5- The Matching

In the matching process, the distance of city blocks was used to determine the absolute difference among two vectors, as in equation (19). The feature vector was obtained by previously mentioned techniques in the proposed system [18].

$$C.B = \sum_{i=1}^n |a_i - b_i| \quad (19)$$

Environment

The suggested method was implemented and tested on a CASIA (V4) database interval. The suggested system was implemented by MATLAB version R2013a program language.

3. RESULTS AND CONCLUSIONS

3.1 Results

System performance

The average time values for feature extraction in the iris region (ROI) for A, B, C, and D, are illustrated in Table-1.

Table 1- Time feature extraction for (ROI)

Iris region	Time
A	31.494814 S
B	10.303709 S
C	1.560958 S
D	1.433534 S

The time for feature extraction of regions A, B, and C was greater than the time elapsed for feature extraction to region (D). Therefore, the system will depend on region (D) to perform the feature extraction which is used in the matching process.

The proposed system used 600 left- and right-eye images from CASIA (V4) database for 30 individuals, each individual has 10 different capture images for the left eye and 10 for the right eye. The sub-region(ROI) of the iris region was used for feature extraction. This region must have equal coordinates (m, n) . To build the DoG pyramid, feature extraction using SIFT needs to detect the number of octaves, levels, sigma, and threshold. Value of octaves=3, level=3, with various values of sigma and threshold were suggested to eliminate low contrast features. The recognition rate for the proposed system was computed as in Eq.20 [8]. Tables 2 and 3 show an increase in the recognition rate due to certain sigma and threshold values imposed. However, at some sigma and threshold values, there were no features extracted, or the recognition rate was decreased. The results in table 2 were obtained using (5) left-eye images for the enrolled stage and (5) left-eye images for the tested stage. Also, region (D) of the iris was used for feature extraction. Table 3 shows the results of (5) right-eye images for the enrolled stage and (5) right-eye images for the tested stage.

$$\text{Recognition Rate} = \frac{\text{Number of correctly identified iris images}}{\text{Total number of iris images}} * 100\% \quad (20)$$

Table 2-The recognition ratio of 300 left-eye images (5 enrolled : 5 tested) for region D by using scale invariant feature transforms and local binary patterns.

Sigma	Threshold	Recognition rate	Number of features	Sigma	Threshold	Recognition rate	Number of features
$\sqrt{2}$	0.1	0%	Not found	$\sqrt{3}$	0.1	0%	Not found
$\sqrt{2}$	0.01	97.33%	128	$\sqrt{3}$	0.01	90.50%	128
$\sqrt{2}$	0.05	98.67%	128	$\sqrt{3}$	0.05	90.55%	128
$\sqrt{2}$	0.07	98.67%	128	$\sqrt{3}$	0.07	92.35%	128
$\sqrt{4}$	0.1	0%	Not found				
$\sqrt{4}$	0.01	93%	128				
$\sqrt{4}$	0.05	91.03%	128				
$\sqrt{4}$	0.07	92.17%	128				

Table 3-The recognition ratio of 300 right-eye image (5 enroll : 5 test) of region D by using scale invariant feature transforms and local binary patterns.

Sigma	Threshold	Recognition rate	Number of features	Sigma	Threshold	Recognition Rate	Number of features
$\sqrt{2}$	0.1	0%	Not found	$\sqrt{3}$	0.1	0%	Not found
$\sqrt{2}$	0.01	92.33%	128	$\sqrt{3}$	0.01	88.66%	128
$\sqrt{2}$	0.05	90.67%	128	$\sqrt{3}$	0.05	85.5%	128
$\sqrt{2}$	0.07	96.66%	128	$\sqrt{3}$	0.07	85.33%	128
$\sqrt{4}$	0.1	0%	Not found				
$\sqrt{4}$	0.01	92%	128				
$\sqrt{4}$	0.05	91.33%	128				
$\sqrt{4}$	0.07	92.67%	128				

- The highest recognition rate, as calculated by Eq.20, was 98.67 for the left-eye and 96.66 for the right-eye, by using SIFT and LBP for the region D, with a value of $\sigma = (\sqrt{2})$ and threshold = 0.07, as illustrate in Figure-12.

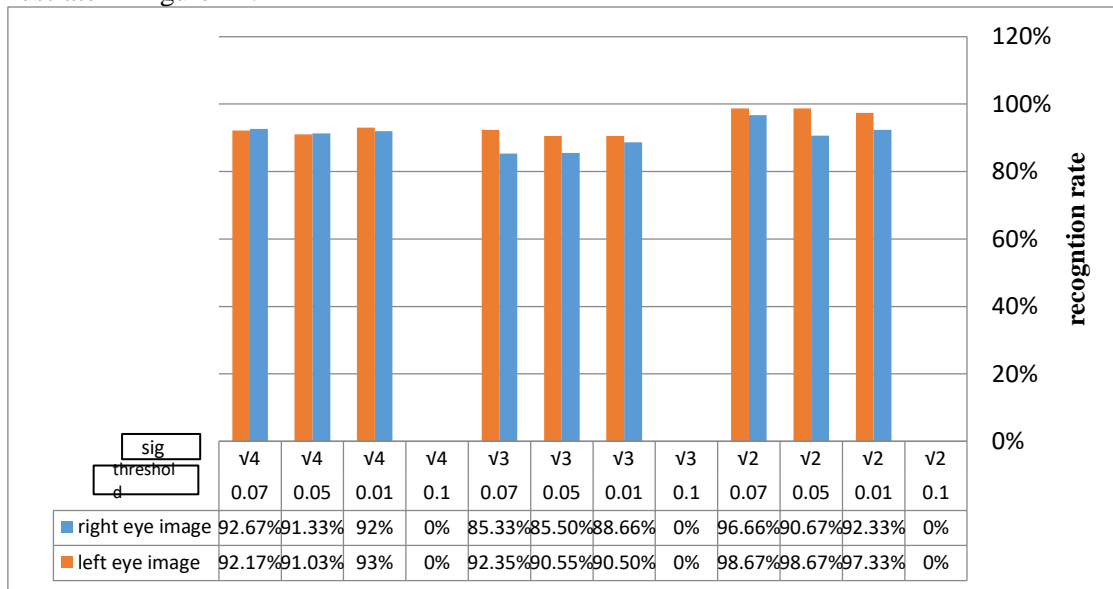


Figure 12- Recognition rate of (5 enrolled: 5 tested) for 30 individuals (left and right eye).

3.2Conclusions

- Resizing of iris images is not useful because it does not eliminate the occlusion caused by eyelids and eyelashes, as illustrated in Figure-10 using ROI
- Taking a large area of iris causes the consumption of more time and storage capacity. Also, it occasionally causes some problems with resources of device (memory, cpu,.. etc.), as explained in Table-1 which shows the computing time of feature extraction, using region D with a size of (100*100)
- Fusion of the SIFT method followed by the LBP method produced the best results with minimal errors when the matching processes was performed, as in Eq.19, between the tested and enrolled images.

References

1. M. Z. Rashad, M. Y. Shams, and O. Nomir, **2011**. "I Rise R Ecognition B Ased on Lbp and," **3(5)**: 43–52, 2011.
2. G. Yang, S. Pang, Y. Yin, Y. Li, and X. Li, **2013**. "SIFT based iris recognition with normalization and enhancement," *Int. J. Mach. Learn. Cybern.* **4(4)**: 401–407.
3. H. Kaur and S. Petunia, **2016**. "Image Enhancement and Iris Recognition using SIFT Feature Extraction," *Int. J. Adv. Res. Electron. Commune. Eng.*, **5(5)**: 1254–1256.
4. C. Rathgeb, J. Wagner, and C. Busch, **2019**. "SIFT-based iris recognition revisited: prerequisites, advantages and improvements," *Pattern Anal. Apple.*, **22(3)**: 889–906.
5. D. A. Roy and U. S. Soni, **2016**. "IRIS segmentation using Daughman’s method," *Int. Conf. Electr. Electron. Optim. Tech. ICEEOT 2016*, pp. 2668–2676.
6. H. K. Rana, M. S. Azam, M. R. Akhtar, J. M. W. Quinn, and M. A. Moni, **2019**. "A fast iris recognition system through optimum feature extraction," *PeerJ Comput. Sci.*, **2019(4)**: 1–13.
7. B. J. A. Al-juburi and H. R. Mohammed, **2017**. "Effect of Irregular Determination on Iris Recognition," **3(2)**: 1–5, 2017.
8. S. A. Ali, **2014**. "Iris Recognition System Based on Texture Features," **6(April)**: 1–12, 2014.

9. B. J. A. Al-Juburi, P. H. Rustum Mohammed, and A. N. H. Al-Shareefi, **2017**. "Iris Recognitions Identification and Verification using Hybrid Techniques," *Res. J. Appl. Sci. Eng. Technol.*, **14**(12): 473–482.
10. S. Singh and K. Singh, **2011**. "Segmentation Techniques for Iris Recognition System," *International Journal of Scientific & Engineering Research*, **2**(4): 1-8.
11. H. Kaur and S. Pathania, **2016**. "Image Enhancement and Iris Recognition using SIFT Feature Extraction," *Int. J. Adv. Res. Electron. Commun. Eng.*, **5**(5): 1254–1256.
12. D. G. Lowe, **2004**. "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, **60**(2): 91–110.
13. K. W. Yang, T. H. Chen, S. X. Xing, and J. X. Li, **2012**. "Infrared and visible image registration base on SIFT features," *Key Eng. Mater.*, **500**: 383–389, 2012.
14. I. Rey Otero and M. Delbracio, 2014. "Anatomy of the SIFT Method," *Image Process. Line*, **4**: 370–396.
15. M. S. Cardoso, **2016**. "Understanding SIFT Algorithm and its uses," **2**(10): 556–560.
16. D. G. Zuo, T. Wen, Z. K. Li, and Z. L. Li, **2014**. "Improved SIFT algorithm based on mixed programming," *Adv. Mater. Res.*, vol. 926–930, pp. 3058–3062.
17. A. Haghnegahdar, S. Kolahi, L. Khojastepour, and F. Tajeripour **2018**. "Diagnosis of Tempromandibular Disorders Using Local Binary Patterns," *Journal of biomedical physics & engineering*, **8**: 87.
18. Y. Sari, M. Alkaff, and R. A. Pramunendar, **2018**. "Iris recognition based on distance similarity and PCA," *AIP Conf. Proc.*, **1977**(June).