

Research Article

An Accurate and Efficient User Authentication Mechanism on Smart Glasses Based on Iris Recognition

Yung-Hui Li and Po-Jen Huang

Department of Computer Science & Information Engineering, National Central University, Taoyuan 32001, Taiwan

Correspondence should be addressed to Yung-Hui Li; yunghui@gmail.com

Received 9 December 2016; Revised 21 March 2017; Accepted 11 April 2017; Published 13 July 2017

Academic Editor: Paolo Bellavista

Copyright © 2017 Yung-Hui Li and Po-Jen Huang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In modern society, mobile devices (such as smart phones and wearable devices) have become indispensable to almost everyone, and people store personal data in devices. Therefore, how to implement user authentication mechanism for private data protection on mobile devices is a very important issue. In this paper, an intelligent iris recognition mechanism is designed to solve the problem of user authentication in wearable smart glasses. Our contributions include hardware and software. On the hardware side, we design a set of internal infrared camera modules, including well-designed infrared light source and lens module, which is able to take clear iris images within 2~5 cm. On the software side, we propose an innovative iris segmentation algorithm which is both efficient and accurate to be used on smart glasses device. Another improvement to the traditional iris recognition is that we propose an intelligent Hamming distance (HD) threshold adaptation method which dynamically fine-tunes the HD threshold used for verification according to empirical data collected. Our final system can perform iris recognition with 66 frames per second on a smart glasses platform with 100% accuracy. As far as we know, this system is the world's first application of iris recognition on smart glasses.

1. Introduction

The advancement of the technology of mobile devices has brought many benefits to the general public. Nowadays, a lot of important tasks can be accomplished using smartphones, such as sending and receiving e-mails, handling business documents, taking photos and videos, and also functions like mobile payment. As technology advances, more and more users' private data is stored on mobile devices. To prevent unauthorized users from accessing devices, it is common for mobile devices to have a user authentication mechanism before it is unlocked, which ensures that only rightful owners can access the private and confidential data stored on the mobile devices.

There are three levels of user authentication mechanisms which can be applied to all security systems. The first level of user authentication mechanism performs user verification based on "what you have?," which asks the user to present a preissued token (e.g., a physical key or NFC card). The second level of user authentication mechanism performs user

verification based on "what you know?," which asks the user to present secret information that only he/she knows for verification purpose (such as password and pattern lock). The third level of user authentication mechanism performs user verification based on "what you are?," which detects and recognizes the user identity based on his/her unique physical characteristics (such as fingerprint, facial images, and iris patterns).

For the traditional authentication mechanism using password, users can set up their own passwords using random combinations of numbers, letters, or special symbols. If the strength of the password is too weak, it can be easily recorded by the shoulder-surfers. On the other hand, if the complexity of the password is high, it is not easy to remember and sometimes it can be easily forgot by user himself.

On Android platform, there is a simplified authentication mechanism called "pattern lock." To unlock the phone, users do not need to remember a set of digital passwords, instead, they have to create a pattern which connects variable number of dots with line segment in a predefined order. However, it

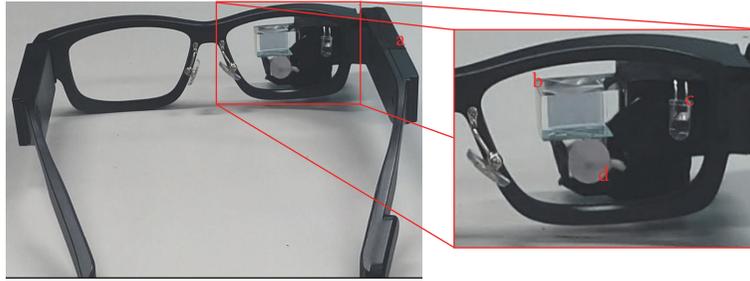


FIGURE 1: Smart glasses with specially designed hardware for iris recognition. (a) Battery module. (b) Micro-projection screen. (c) Infrared LED illuminator. (d) IR camera module.

is a common phenomenon that the perspiration and grease attached on human fingers leave a visible sliding mark on the screen, revealing the sliding patterns to unlawful users.

For the case of the wearable smart glasses, there is no keyboard or touch screen for users to interact with the system. Therefore, it is virtually impossible to ask users to either enter passwords or unlock the device with sliding patterns. Currently there is no feasible user authentication mechanism on such platform which offers high security and convenience at the same time. Such authentication mechanism is urgently needed. Biometric recognition can be a feasible solution to this problem. Among all possible modalities in biometrics, fingerprint is the one with the highest visibility on the smartphones. However, fingerprint recognition is vulnerable to the attack due to the recent emergence of 3D printing technology.

In the 2014 Chaos Computer Club, which is Europe's largest association of hackers, a security officer named Starbug demonstrated how to copy the German Defense Secretary's fingerprints by first capturing resolving high-resolution palm photos and then printing the fake fingerprint using 3D printing technology [1]. Such demonstration shows that there is a possibility of circumventing the user authentication mechanism protected by fingerprint recognition.

In this paper, we propose a feasible solution for efficient and accurate user authentication for wearable smart glasses devices. We argue that iris recognition is the most suitable biometric modality for this purpose, and we implement such mechanism on a commercial smart glasses from scratch, as shown in Figure 1. Our contribution includes achievement in both hardware and software.

For the hardware side, we redesigned the original smart glasses hardware: a new infrared (IR) photographic photosensitive module, which contains well-designed infrared light source, IR imaging sensor, and the fine-tuned lens module, which precisely focuses on the iris area when the user wears the glasses. The specifications of the original smart glasses are as shown in Table 1. Table 2 lists the specifications for IR sensors and illuminators.

On the software side, we use Android Studio to develop an Android program to perform iris recognition. Our implementation follows the framework proposed by Daugman [2], with a brand new redesigned algorithm for the task of iris segmentation. Based on the integration of hardware and

TABLE 1: The specification for iris camera module.

Light source	IR LED dimming group, 850 nm 90 mW TSHG6410 VISHAY IR LED, 34° lens can cast light at 3 cm distance
Input voltage	DC 5 V
Size	44.7 * 64.7 * 31.8
Sensor	CMOS

TABLE 2: Smart glasses hardware specifications.

CPU	Dual core Cortex-A9, 1.2 GHz
HDD	8 GB NAND Flash + 1 GB DDR3 SDRAM
Battery	Li Battery 610 mA/2.2 Wh/3.7 V
OS	Android 4.4.2
Camera	Color CMOS 5 megapixel sensor 0.29 inch LCoS Module, Aspect Ratio: 15 : 9,
Monitor	See through type Brightness: 700 cd/m@0.3 W (3 V/100 mA), Resolution: 800 × 480 (WVGA), Image size: at 3 m see 30'' virtual screen

software specially designed for smart glasses platform, we are able to successfully implement the user authentication mechanism based on the iris recognition technology.

2. Literature Review

The earliest and most popular iris recognition algorithm was established by Professor Daugman [2]. According to his work [2], iris recognition is performed through the following stages of iris image capturing, image preprocessing, iris segmentation, iris image normalization, feature extraction, and feature matching, respectively, as shown in Figure 2.

Among these stages, the goal of image preprocessing is to remove the noise and make the iris feature more prominent. The goal of iris segmentation stage is to localize the iris region by recovering the inner and outer boundary of the iris. The stage of iris normalization performs an image transformation from Cartesian to Polar coordinate. The feature extraction stage extracts iris image features using specially designed filters like Gabor or other wavelets and then quantizes the complex valued features into binary features. Finally, the

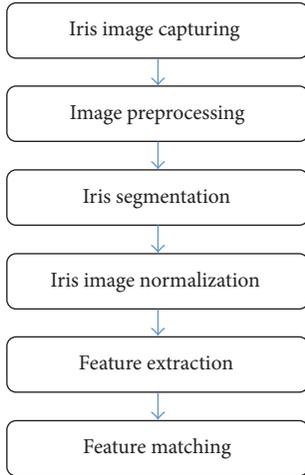


FIGURE 2: The flowchart of a typical iris recognition system, based on Daugman's frameworks [2].

feature matching stage calculates a distance (which is called Hamming distance, HD) between two iris features by using exclusive OR operation. An adequate threshold can be set for HD in order to perform verification.

Among all stages, the accuracy of iris segmentation stage is crucial to the iris recognition rate. It is obvious that if the iris boundaries are not localized correctly, the feature extraction and matching would be based on areas which are not iris textures, and therefore the recognition rate drops dramatically. Early works like Daugman's algorithms [2–4] assume that both pupil and iris have circular form and apply integrodifferential operators to look for the maxima value in the blurred partial derivative and thus recover the iris and pupil contour, as shown in

$$\max_{(r, x_0, y_0)} \left| G_\sigma(r) * \frac{\partial}{\partial r} \oint_{r, x_0, y_0} \frac{I(x, y)}{2\pi r} ds \right|. \quad (1)$$

Wildes proposed to apply circular Hough transform to the edge map of the eye image for the purpose of iris segmentation [5]. A more recent work of Daugman suggests using active contour for iris segmentation in order to deal with noncircular boundaries and enable flexible coordinate system [6].

The abovementioned method achieves high accuracy with the cost of high computational complexity. This is not feasible on mobile device, because, for consumer electronic devices, users expect the device to be responsive in all circumstances, especially when performing user authentication.

How to extract iris feature is another influential factor of the iris recognition system, which can be performed by calculating zero-crossings of the wavelet transform at various resolution levels on the concentric circle of the iris to get the resulting one-dimensional (1D) signals [7] or using 2D Gabor filters [2–5], 2D wavelet transform [8], and dyadic wavelet [9] to extract iris feature. The different proposed methods have their own advantages.

There are evolving user authentication mechanisms implemented on mobile device as new techniques emerge

every day, including using electrocardiogram (ECG) signals collected from mobile device for user authentication [10, 11], user authentication based on gait, face, keystroke, or voice measured by the mobile sensors [12, 13], and user authentication based on combinations of multiple mechanism [14, 15]. However, the system with continuous data collection or combination with multiple sensors needs high power durability and it is not practically useful for mobile devices.

In addition, there are some improvement in both pass-words mechanism [16–18] and gestures-based methods [19, 20]. However, for the smart glasses device, even for the model which comes with a touchpad, the area of the touchpad is, most of the time, too small for the users to input complex sliding patterns or long password for user authentication.

3. Proposed Method

3.1. New Hardware Design. Infrared illumination can make the iris border more apparent, because the infrared absorption of the iris and pupil is high, while the sclera reflects most of the infrared light. Based on the reference, the standards proposed from ICNIRP [21], and considering the distance between camera module and the eyes, we choose the infrared light bulb with 850 nm peak wavelength and 90 mW radiation intensity, in order to protect the cornea and the lens of the eye. At normal temperature, the exposure limit for the eyes is 100 mW/cm² within 45 seconds and 320 mW/cm² within 10 seconds. In our application, the IR bulb is enabled only on the stage of “iris image capturing,” and the whole process will be completed in about three seconds, which is much less than the specification listed above (45 seconds and 10 seconds). In addition, the normal iris radius is 0.6 cm, and, on the smart glasses device, the distance between the eyes and light source is about 2 cm. For the purpose of making IR light evenly spread across all of the iris area, the scattering angle of the IR bulb should be at least 16.71° (as illustrated in Figure 3).

Figure 4 is a screenshot from our iris recognition APP which serves two tasks for iris recognition. One is the “Sign-up” process when the user wants to register a new iris template, and the other is the “Login” process which verifies user's identity. The flowchart of these two tasks is shown in Figure 5.

3.2. Efficient Iris Segmentation Algorithm for Mobile Devices.

When we capture the eye images with the IR illumination, we can roughly estimate the possible location of the pupil by first locating the specular reflection caused by IR LED. To find out the coordinates of specularity, we trace out each row and column on the image and mark them if there exists a pixel value larger than 250 in the row or column. Finally, the median of the rows and columns will be set to the coordinates of the specular point. After repeated experiments, we find that the distance value between the coordinates of the specular point and the center of pupil is close to a constant value. Therefore, we are able to roughly estimate a point P in pupillary area. In order to precisely recover the pupillary boundary, we perform a column scan at the x coordinate of P . A set of predefined thresholds ($th_N = 40$, $th_S = 40$, $th_W = 40$, and $th_E = 15$) of pixel intensity is used to select

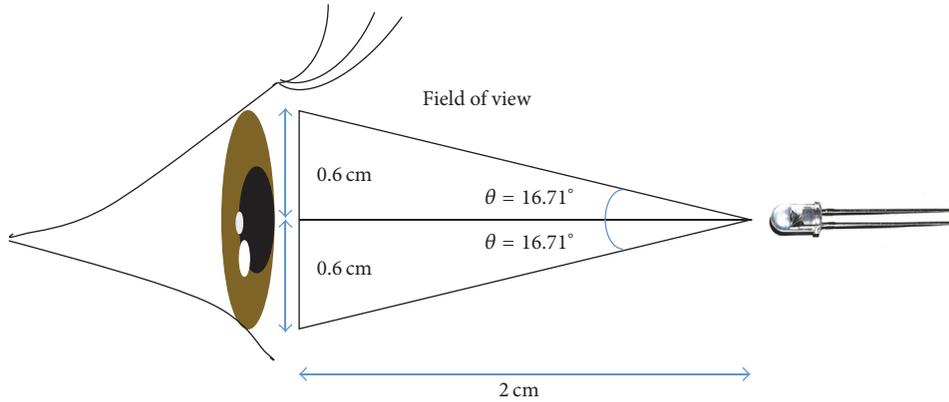


FIGURE 3: The use case emulation for the IR illuminator used on a smart glass. Based on the distance between the IR LED and the eye, and the radius of the eyeball, the scattering angle of the IR illuminator is set to be 16.71°.

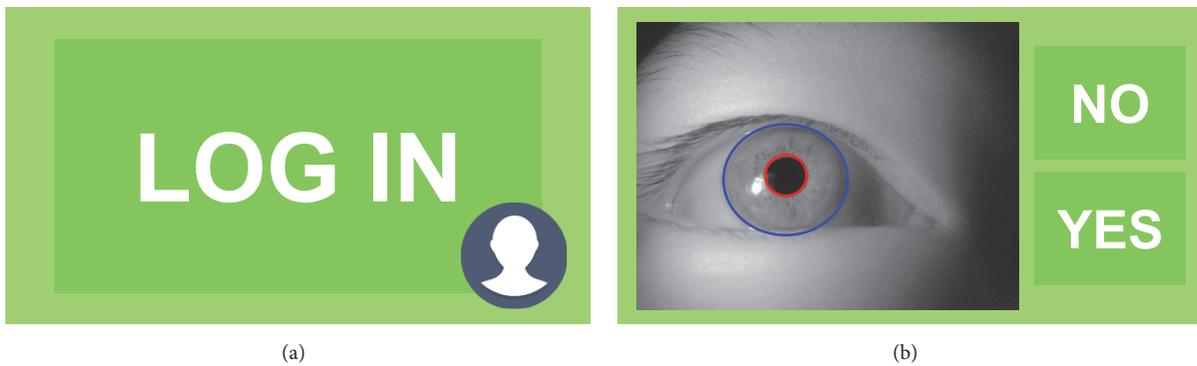


FIGURE 4: Screenshot of our iris recognition APP. (a) User login/sign-up stage. (b) Iris image capturing stage.

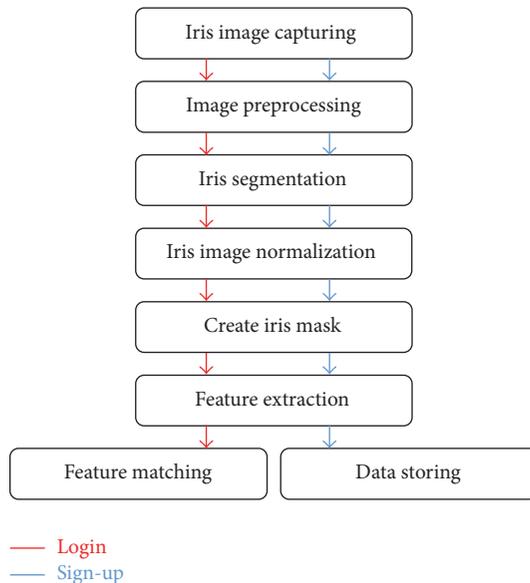


FIGURE 5: Flowchart of our iris recognition program. There are two tasks flows represented in the same chart: “Login” and “Sign-up,” denoted by the red and blue arrows, respectively.

the upper and lower point corresponding to the pupillary boundary on the column. A row scan is performed in the same manner to recover the left- and right end-point of the pupillary boundary on the row where P locates. Thus, we are

able to recover four key points of pupillary region based on P . Finally, the circle fitted through these four key points is the approximation of the pupillary boundary, as illustrated in Figure 6.

- (1) Generate a few emitting rays that go from iris region to sclera region with fine-tuned parameter set $\{s_1, s_2, \theta\}$.
- (2) For each emitting ray locate the position that exhibits the maximal value of the intensity gradient. These positions correspond to the hypothesis of the boundary point set.
- (3) Given the hypothesis of the boundary point set fit a parameterized shape (circle or ellipse) to recover the outer boundary of an iris.

ALGORITHM 1: The procedure of MIGREP.

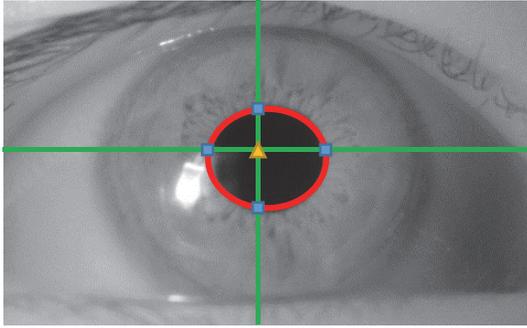


FIGURE 6: Illustration of proposed method for pupil localization. The yellow triangle mark indicates point P , and the blue rectangle marks indicate four key points.

We estimate the iris boundary after the pupillary region and its boundary is localized. Such sequential order simplifies the problem since the outer boundary always resides outside the pupillary region. We proposed a method called MIGREP (Maximization of the Intensity Gradient along the Radial Emitting Path) for outer boundary localization. The first step of MIGREP is to design the path of a few radially emitting rays that go outward from pupillary center. Since we already know the location of the pupillary boundary, the distance between the starting points of the emitting rays and the pupil center can be set to a value s_1 , whose default value is greater than the pupillary radius (the default value is $s_1 = \text{pupillary radius} + 20$). The distance between the end points of the emitting rays and the pupil center should be set to another value s_2 ($s_2 = \text{pupillary radius} + 80$) which makes the end points fall into the sclera region. In this way, most of the emitting rays are supposed to start from somewhere inside iris region and stop at somewhere in the sclera region. By keeping records of the pixel intensity values along the emitting ray, we can locate the position that exhibits the maximal variation of pixel intensity. This position should correspond to the intersection between the emitting ray and the iris boundary. Thus, we can successfully estimate multiple boundary points if multiple emitting paths are employed. The procedure of MIGREP is summarized in Algorithm 1. A pictorial example of MIGREP is given in Figure 7.

3.3. Boundary Point Selection Algorithms for Accurate Iris Boundary Localization. At step (2) of MIGREP algorithm, depending on the parameter θ and the shape of the eyelids

and eyelashes, it is highly possible that the position that exhibits the maximal value of the intensity gradient is not located on the iris boundary, as shown in Figure 8.

To solve this problem, we do not need to consider only a single point where global maximal gradient occurs. Instead, we should consider a set of candidate points where local maximal gradient occurs. For example, in Figure 8, we should consider a set of candidate points consisting of red point and yellow point and then pick the one from the set with the highest likelihood.

We developed a more sophisticated boundary point selection algorithm for this problem. Figure 9 illustrates the idea. First, ten emitting rays are drawn with parameter $\theta \in [173^\circ, 200^\circ]$. For rays with such angles, it is highly likely that maximal gradient occurs on the iris boundary, as shown in Figure 9(a). Thus, the median distance d from these points to the pupillary center is recorded. Second, a new emitting ray (with $\theta \in [140^\circ, 170^\circ]$) is drawn, for which it is likely that incorrect boundary point happens to have the maximal gradient, as shown in Figures 9(b)–9(e). For such case, we consider all points where local maximal gradient occurs and record the corresponding radius value between these points and the pupillary center. Take Figure 8 as an example; the set $\{r_1, r_2\}$ will be recorded. Then, the candidate point is selected based on

$$k = \arg \min_i (d - r_i)^2. \quad (2)$$

Take Figure 9(b) as an example; the red point on the new line will be selected based on (2), instead of the yellow point. Third, after the best candidate point is picked, we update the value d with r_k , which serves as a new approximate value of the radius for boundary points that are close to it. Then we draw next emitting ray with a new θ value and repeat the above mechanism for boundary point selection and distance updating. Repeating the above procedure, we are able to locate many boundary points with θ values in the range of $[-60^\circ, 50^\circ]$ and $[130^\circ, 240^\circ]$, as shown in Figures 9(b)–9(f). As it is shown, if multiple boundary points with wide angle variation can be correctly recovered, the iris boundary can be accurately localized by fitting a circle on all of these boundary points. Algorithm 2 describes the procedure of the boundary point selection algorithm.

3.4. Iris Normalization on Smart Glasses. After successful iris segmentation, we get the parameters of the pupil circle (x_1, y_1, r_1) and the iris circle (x_2, y_2, r_2) . To normalize the iris image, we can partition the inner and outer circumferences into the n equal segments. Assume the endpoints of these segments are labeled B_j and A_j , we can again partition the vector $\overline{A_j B_j}$ into m equal parts, as shown in Figure 10. The coordinates of these endpoints are

$$\begin{aligned} x_{ji} &= x_{A_j} + \frac{x_{B_j} - x_{A_j}}{m} \times i, \\ y_{ji} &= \frac{y_{B_j} - y_{A_j}}{x_{B_j} - x_{A_j}} (x_{ji} - x_{B_j}) + y_{B_j}, \end{aligned} \quad (3)$$

$$1 \leq i \leq m, 1 \leq j \leq n,$$

- (1) Use ten emitting rays with $\theta \in [173^\circ, 200^\circ]$ (for the other side $\theta \in [-4^\circ, -31^\circ]$) to recover the boundary points for these angles by selecting points with maximal gradient on the rays.
- (2) Determine distance threshold d by taking the median value of the distance from these points to the pupillary center.
- (3) Repeatedly do the following steps for a set of θ values even spaced within $[130^\circ, 240^\circ]$ ($[-60^\circ, 50^\circ]$):
 - (A) Draw a new emitting ray based on θ
 - (B) Collect a set of candidate points based on local maximal gradient. From this set, create a set of corresponding radiuses
 - (C) Select the boundary point based on the equation:

$$k = \arg \min_i (d - r_i)^2$$
 - (D) Update d with r_k

ALGORITHM 2: Procedure for boundary point selection algorithm.

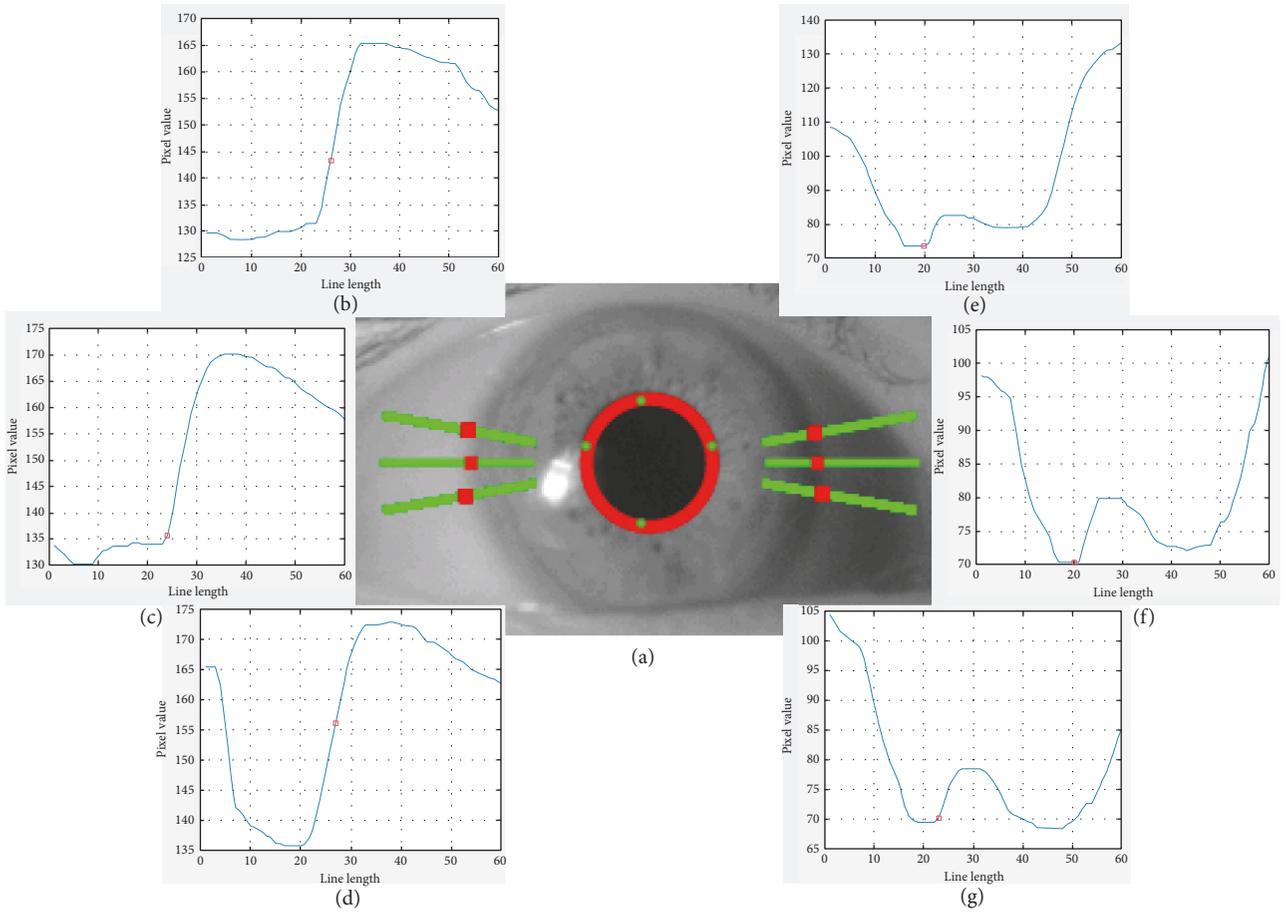


FIGURE 7: Illustration of the proposed procedure (MIGREP) for outer boundary localization. (a) The input eye image and six radial emitting paths (plotted in green) along different angles. (b)~(g) The pixel intensity plots along the corresponding green path shown in (a). The red square denotes the position where the maximal gradient of intensity occurs. Those positions correspond to the intersection points between the outer boundary and the radial emitting path.

where the coordinates of B_j are defined as

$$\begin{aligned} x_{B_j} &= x_1 + \cos \frac{2\pi j}{n}, \\ y_{B_j} &= y_1 + \sin \frac{2\pi j}{n}, \end{aligned} \quad (4)$$

$1 \leq j \leq n.$

And the coordinates of A_j are defined as

$$\begin{aligned} x_{A_j} &= x_2 + \cos \frac{2\pi j}{n}, \\ y_{A_j} &= y_2 + \sin \frac{2\pi j}{n}, \end{aligned} \quad (5)$$

$1 \leq j \leq n.$

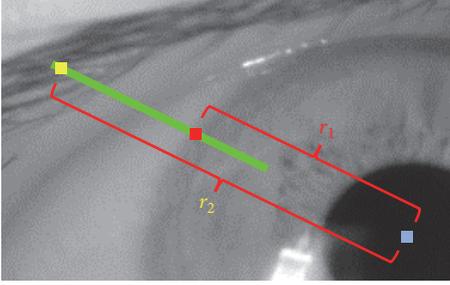


FIGURE 8: Illustration of the problem of boundary point selection. At step (2) of MIGREP algorithm, it is possible that the gradient value of yellow point is higher than that of red point, which is an incorrect boundary point estimation.

By the above equation, the pixel values on the original image can be assigned to the corresponding position on a rectangle of size $m \times n$, and the iris normalization can be done in an efficient way.

3.5. Iris Mask Estimation. Before extracting the feature in polar domain, it is important to create an accurate iris mask in order to preclude the noniris region from feature extraction and matching, as shown in Figure 11. In this work, we propose to use two mask estimation methods. The first method we used is the rule-based method which is similar to what is described in Kong and Zhang’s work [25]. It basically detects whether there is a strong variance of pixel intensity in a local window and uses it as a feature for classification. The second method we applied is a machine learning based iris occlusion estimation algorithm, which is based on the Gabor filter response and Gaussian Mixture Models (GMM). The details of the algorithm are described in [26–29]. It has been proved to be a robust and accurate method for iris mask generation.

3.6. Iris Features and Matching Algorithms. The feature we used for iris feature extraction is the Haar-based wavelet feature, similar to the method proposed in [30].

Another iris matching method we propose to use in this work is a patch-based probabilistic graphical model (PGM) method. It divides the whole iris image into local patches and models the local deformation with PGM. Its recognition accuracy has been shown to outperform the classical Daugman’s method, as described in [23, 24]. Using PGM, we expect to see higher recognition accuracy in a low quality iris database.

3.7. Intelligent HD Threshold Adaptation. In our preliminary experiment, it is observed that though the HD distribution of authentic and imposter matching is totally disjoint for all subjects, the optimal threshold for separating these two distributions varies for different users. Such phenomenon is shown in Table 3.

In this work, since the goal is to perform user authentication, which is a one-to-one matching scenario, we propose a new method for iris matching stage. Traditionally, in Daugman’s framework, a predefined threshold T of HD is set

TABLE 3: The HD distribution of each subject. The column of “correct” and “incorrect” lists the number of correct or incorrect verification based on a fixed threshold.

	# correct	# incorrect	HD (authentic)	HD (imposter)
Person 1	9	1	<0.153	>0.335
Person 2	9	0	<0.206	>0.344
Person 3	7	0	<0.263	>0.342
Person 4	10	0	<0.232	>0.370
Person 5	8	2	<0.350	>0.407
Person 6	10	0	<0.253	>0.360
Person 7	4	6	<0.354	>0.367
Person 8	8	0	<0.213	>0.354
Person 9	10	0	<0.179	>0.411
Person 10	5	5	<0.367	>0.402
Person 11	10	0	<0.239	>0.359
Person 12	8	0	<0.214	>0.427
Person 13	10	0	<0.192	>0.375
Person 14	9	1	<0.245	>0.312
Person 15	7	0	<0.220	>0.382

for iris feature matching stage. When the HD is smaller than T we assert that these two irises match each other; otherwise it is a nonmatch. In this work, we propose an intelligent HD threshold adaptation method which intelligently adjusts T to adapt to the HD distribution that is commonly observed for target user.

Specifically speaking, for each target user, we collect N feature matching results (N HD values), $N/2$ of which comes from authentic matching and the other $N/2$ from imposter matching. N is a value which is freely chosen by the programmer. In our experiment, $N = 10$.

Then each HD distribution (authentic and imposter matching score) is fitted with a single Gaussian distribution. The optimal HD threshold is determined by locating the intersection of the two Gaussian distributions.

4. Experimental Results

4.1. Iris Segmentation. We collect eye images from 62 subjects, with about 15 pictures of each subject. The subjects are asked to wear the smart glasses with their naked eyes and images of their right-eye were taken using the specially designed IR camera module attached in the smart glasses.

The blink, squint, and blurred images are removed from the collected data, and the remaining 10 images are selected as experimental data. Finally, our dataset has accumulated a total of 620 eye images. When an eye image is loaded into the iris recognition program of the smart glasses, its resolution will be resized into 480×360 ; then the program will detect the inner and outer boundaries of the iris in the image automatically and display red and blue circle for contours, respectively. To determine iris segmentation performance, the contours detected by the program will be compared to manually created contours (ground-truth). The image will be labeled as “correct” if the contour estimated by the program approximates the actual iris boundary with little or no errors;

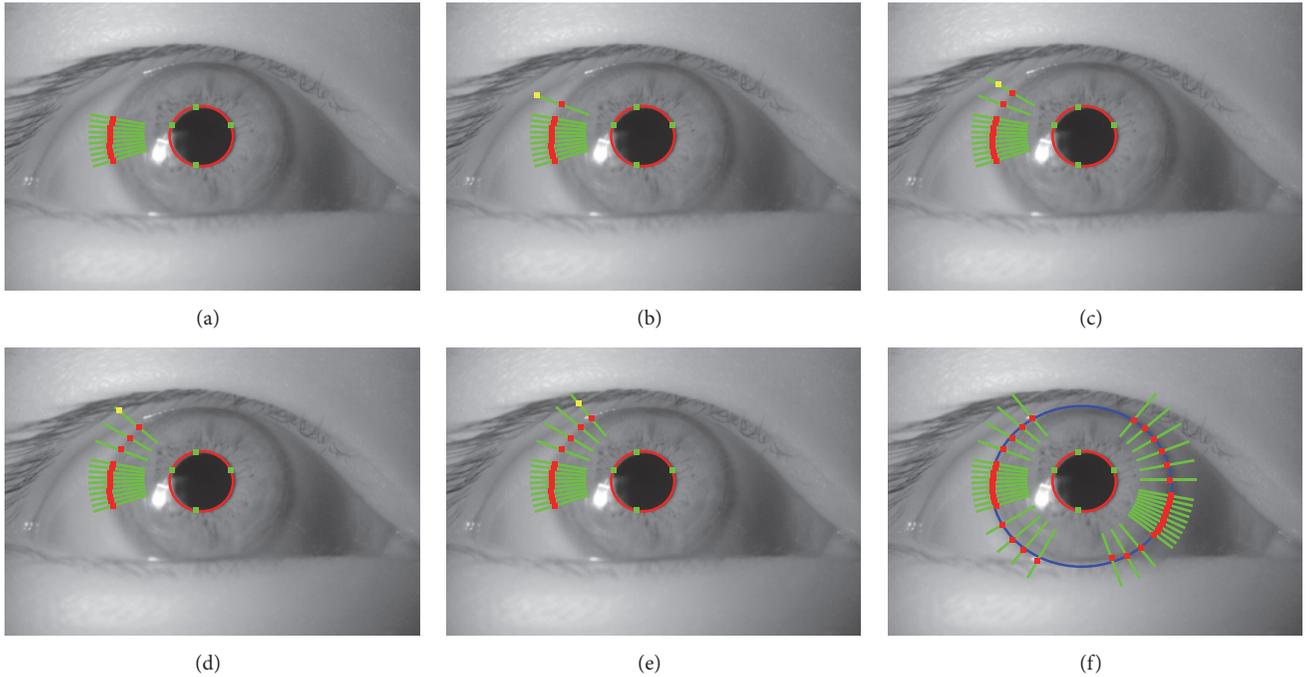


FIGURE 9: Illustration of boundary point selection algorithm for step (2) in MIGREP algorithm. (a) Ten emitting rays are drawn with parameter $\theta \in [173^\circ, 200^\circ]$, and the points corresponding to the maximal gradient are recorded. A distance threshold value d is determined by taking the median of the distance from these points to the pupillary center. (b) A new emitting ray is drawn with larger angle θ . Applying the boundary point selection algorithm described in Algorithm 2, we can correctly select red point as new boundary point. (c)–(e) Repeatedly drawing new emitting rays and applying the boundary point selection algorithm, we are able to locate many boundary points correctly. (f) Final iris boundary is recovered by fitting a circle on all of the recovered boundary points.

TABLE 4: The time cost for iris segmentation of both algorithms.

Algorithm	Average (time cost)	Std (time cost)	Total (time cost)
Baseline	7.4778 (s)	0.9926 (s)	4636.238 (s)
Proposed	0.3052 (s)	0.0635 (s)	189.215 (s)

otherwise it will be labeled as “incorrect” if there is a visible difference between the estimated boundary and the ground-truth boundary.

For comparison purpose, we implement the algorithm proposed in [22] as the baseline to perform large-scale iris recognition in our dataset and compare their performance with our methods. To compare the efficiency of both algorithms, we record the cost of time and results of segmentation for each image. The statistics are listed in Table 4. It shows that the time our algorithm spent for iris segmentation on the dataset is far less than the baseline algorithm. In addition, Figure 12 shows the time cost for segmenting each image in dataset, which shows that our algorithm has a high degree of stability of the processing time. There are 611 images among 620 images which can be correctly segmented by the proposed algorithm. The successful rate is 98.54%. On the other hand, the successful rate of the baseline algorithm is 74.82%. Examples of segmentation hypotheses for both algorithms are shown in Figure 13.

TABLE 5: The performance of iris segmentation on MICHE database.

Device	Environment	# images	Successful rate	
			Proposed	Baseline
Apple iPhone 5	Indoors	631	96.83%	9.19%
Samsung Galaxy S4	Indoors	663	93.51%	8.74%
Samsung Tablet II	Indoors	316	97.78%	17.08%

We also performed experiments on MICHE database [31], which is a public available iris database collected using popular mobile devices, including Apple iPhone 5, Samsung Galaxy S4, and Samsung Tablet II. Since the main purpose of our study in this paper is to propose a new and effective iris recognition algorithm on mobile devices, performing experiments on MICHE database is more adequate for fair evaluation of the performance of the algorithms. Table 5 lists the comparisons of the iris segmentation accuracy between the proposed method and the baseline method [22].

4.2. Iris Recognition Results on Self-Collected Database. The large-scale iris recognition experiments are performed using the polar-domain iris images derived in iris segmentation stage. Since we have implemented two iris segmentation algorithms (the baseline method [22] and the proposed method),

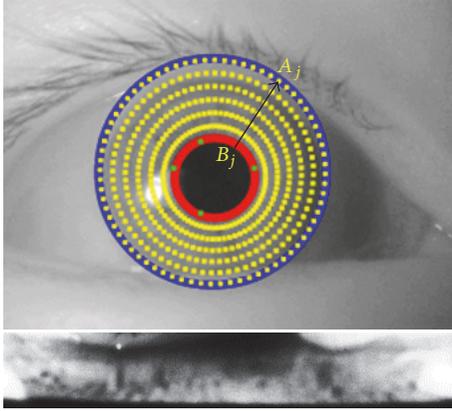


FIGURE 10: Illustration of iris image normalization. Upper: the partitioning of the iris region in Cartesian coordinate. Lower: the normalized iris image in polar coordinate.



FIGURE 11: Illustration of iris mask.

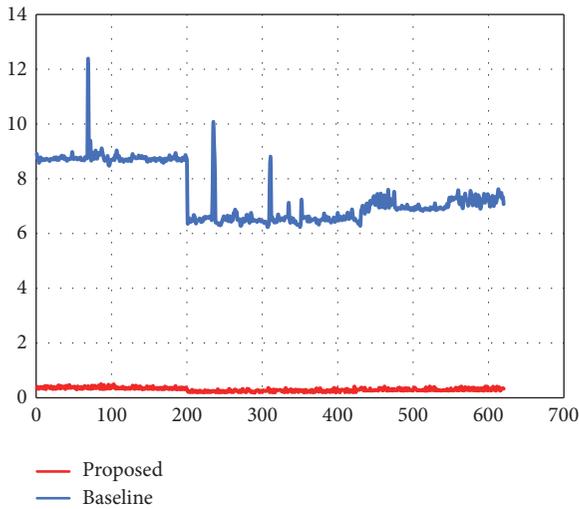


FIGURE 12: Illustration of the time cost of each image in dataset. The red and blue curve denote the required time for iris segmentation using the proposed and the baseline algorithm, respectively.

there will be two sets of polar-domain iris images. All images, including both correctly and incorrectly segmented iris images, are used in iris recognition experiments.

For the iris recognition experiments, according to the description in Section 3.6, we implemented two iris matching schemes: HD-based and PGM-based method [23, 24]. According to the variation of the iris mask estimation algorithms, the HD-based method can be further divided into (a) rule-based iris mask estimation [25] and (b) GMM-based iris mask estimation [26–29]. For PGM-based method, since the iris mask is directly estimated inside the graphical model, there is no need to consider other variations.

Figure 14 shows the large-scale iris recognition results, where each subplot shows the ROC curve of the results when using (a) rule-based iris mask and Haar-based iris features (RB-HD); (b) GMM-based iris mask and Haar-based iris features (GMM-HD); (c) PGM-based method. Figure 15 shows the Equal Error Rate (EER) of the iris recognition results based on the three iris recognition schemes.

From Figures 14 and 15, we can see that no matter which iris matching algorithms are being used, the polar-domain iris images segmented with the proposed method can achieve higher recognition accuracy, compared to segmentation with the baseline method. The results measured from EER values are aligned with ROC trends. They demonstrate that the proposed method can achieve high accuracy in iris segmentation as well as iris recognition.

As stated in Section 3.7, when performing user authentication (which is 1:1 matching), we can further apply the method of HD threshold adaptation. After applying HD threshold adaptation method with the iris matching algorithm RB-HD, the user authentication rate achieves to 100%, with 0% FAR and 0% FRR.

4.3. Iris Recognition Results on MICHE Database. As stated in Section 4.1, besides using self-collected database, we also performed iris recognition experiments on MICHE, which is a publicly available database. Similar to the procedure stated in Section 4.2, we used the polar-domain iris images (segmented using the proposed method) for all three subsets of MICHE in this experiment. The three subsets are denoted as IP5 (for iPhone 5), GS4 (for Samsung Galaxy S4), and GT2 (for Samsung Tablet II), respectively. The iris recognition algorithms we used in this section is the same sets of algorithms described in Section 4.2, including RB-HD, GMM-HD, and PGM.

Since there are totally three datasets and three iris recognition algorithms being compared, the number of all combinations is nine. The ROC curves of the iris recognition results for each of the three subsets are shown in Figure 16. The EER values of all nine combinations are shown in Figure 17.

4.4. Efficiency of the Proposed System. The experimental results reported in Sections 4.1–4.3 are executed on a personal computer, coded with MATLAB R2016b. In order to test the efficiency and practicality of the proposed system, we further implemented the whole proposed iris recognition method on a mobile computing platform by rewriting all codes with QT/OpenCV. On a mobile platform with Cortex-A9 CPU, it is able to achieve to the speed of 66.7 frames per second (fps). Table 6 lists the time it requires to perform the task in each stage of iris recognition on Cortex-A9 CPU. It only takes 15 ms to process a single iris image, which is more than enough for the use of user authentication on mobile devices.

5. Discussion and Conclusion

In this work, we proposed a solution for implementing iris recognition mechanism on a wearable smart glasses

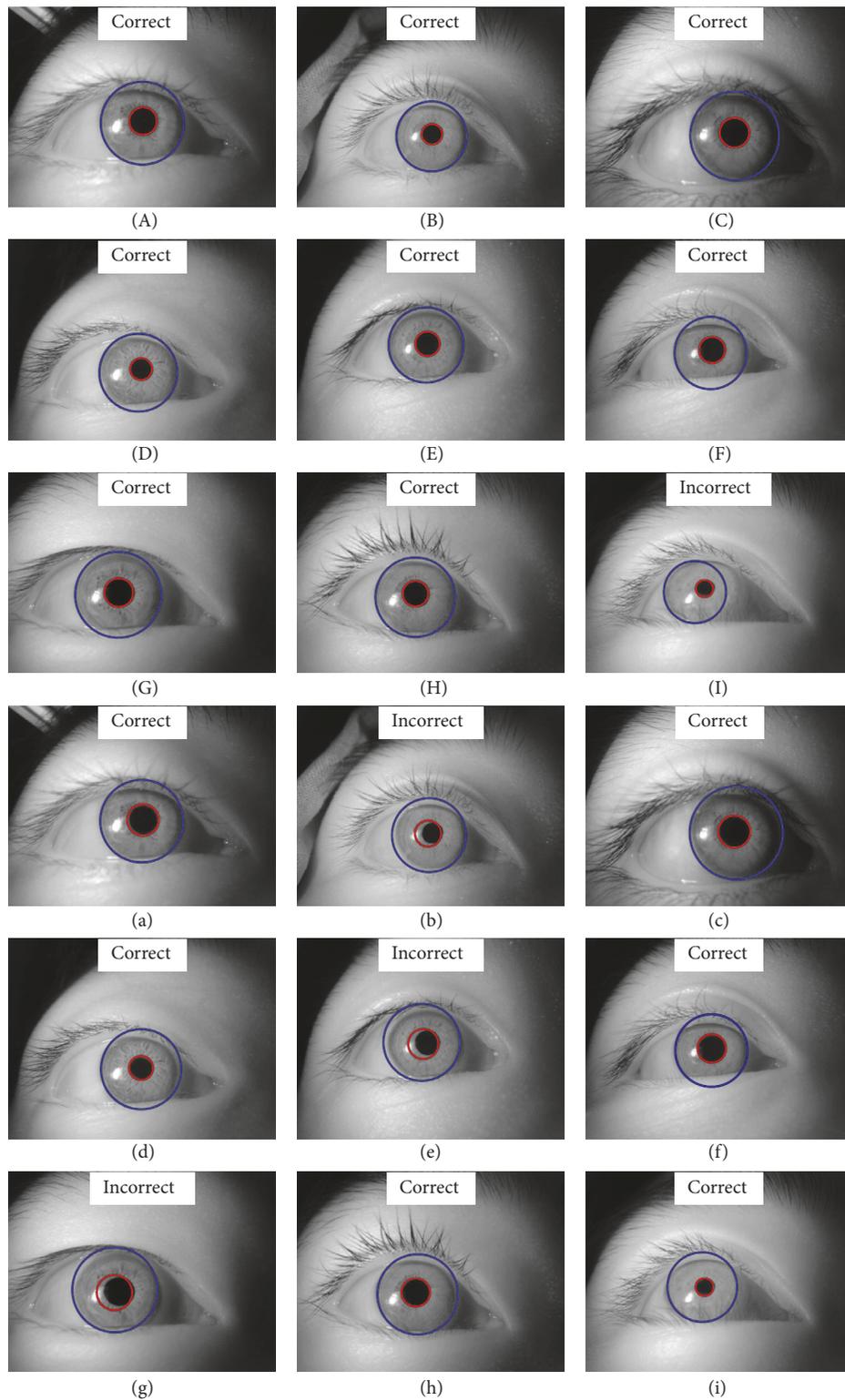


FIGURE 13: The estimated inner and outer iris boundaries. Upper nine illustrations: (A)~(I) are the segmentation results using proposed method. Lower nine illustrations: (a)~(i) are the segmentation results using method [22]. The same alphabet pairs (i.e., (A) and (a)) represent the same input image, while uppercase and lowercase denote the fact that the results derived from the proposed algorithm and the baseline algorithm, respectively.

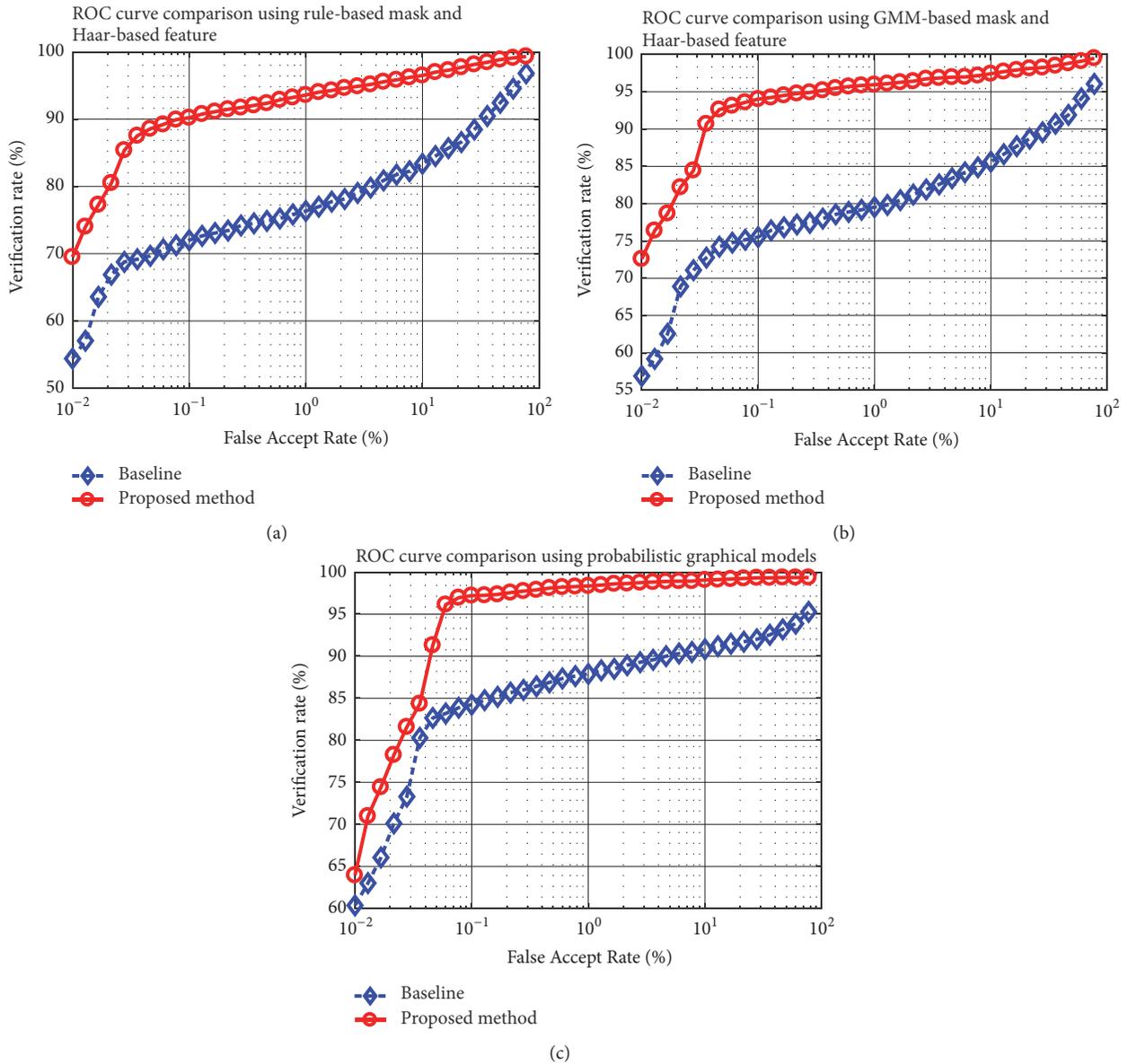


FIGURE 14: Large-scale iris recognition performance comparison between the proposed method and the baseline method [22], on self-collected database. These are ROC curves of both methods when using (a) rule-based iris mask and Haar-based iris features (RB-HD); (b) GMM-based iris mask and Haar-based iris features (GMM-HD); (c) iris matching with probabilistic graphical models (PGM) [23, 24].

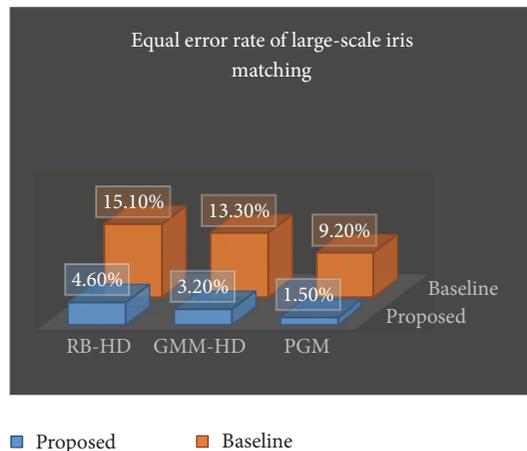


FIGURE 15: The Equal Error Rate (EER) of the iris recognition experimental results on self-collected database with three iris matching algorithms, which are (a) RB-HD: rule-based iris masks and Haar-based HD; (b) GMM-HD: GMM-based iris masks and Haar-based HD; (c) PGM: probabilistic graphical model based iris matching.

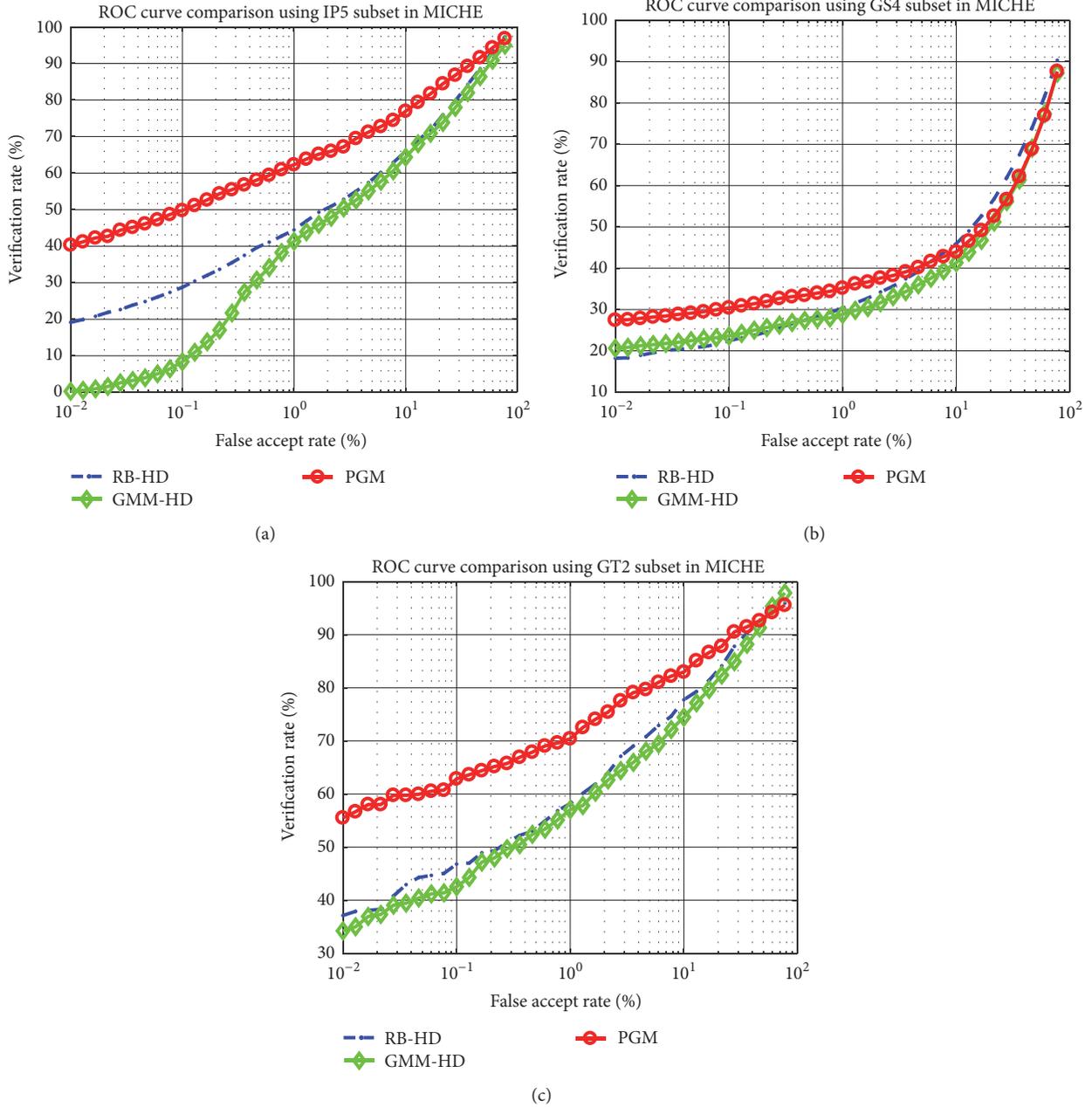


FIGURE 16: Large-scale iris recognition performance comparison using MICHE database: (a) for IP5 subset; (b) for GS4 subset; (c) for GT2 subset.

device. The solution includes specially crafted hardware and software for such purpose. On hardware side, a set of internal infrared camera modules, including well-designed infrared light source and lens module, is carefully designed which is able to take clear iris images within 2~5 cm.

On software side, we have two contributions. First, we propose a new iris segmentation algorithm, which is very robust toward environmental lighting and eyelids/eyelash occlusion pattern variations. Another advantage of our algorithm is its efficiency. If implemented correctly in C++ level, it only takes 10 ms to localize one iris image on a Cortex-A9 platform.

The second contribution is that we propose to use an intelligent HD threshold adaptation method during the feature matching stage. Such method is able to determine the best HD threshold for different subject, according to the accumulated HD matching records. Hence, different HD threshold can be applied to different subject, which enhances the recognition performance and decreases the probability of false acceptance.

Our solution is able to execute on a Cortex-A9 platform with speed of 66 fps. In our experiments, 98.54% of iris images can be successfully localized. For those images, the verification rate achieves 100%. It shows that our proposed

TABLE 6: The average time it takes to complete the task in each stage of iris recognition on Cortex-A9 CPU.

Stage	Average time
Iris segmentation	10 (ms)
Iris normalization	2 (ms)
Iris mask generation	1 (ms)
Feature extraction	2 (ms)
Feature matching	0.001 (ms)
Total	15.001 (ms)

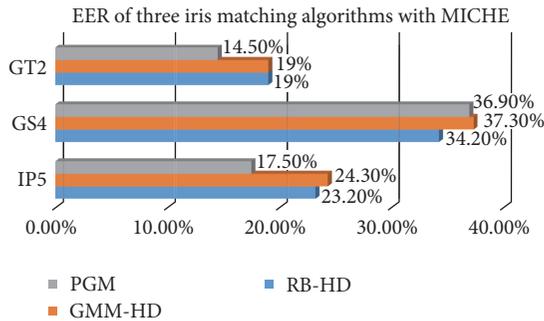


FIGURE 17: The Equal Error Rate (EER) of the iris recognition experimental results on MICHE, with three iris matching algorithms, which are (a) RB-HD: rule-based iris masks and Haar-based HD; (b) GMM-HD: GMM-based iris masks and Haar-based HD; (c) PGM: probabilistic graphical model based iris matching.

solution is both efficient (in terms of speed) and effective (in terms of accuracy). To the best of the authors' knowledge, this is the first published work about implementing iris recognition mechanism on smart glasses platforms.

The future work consists of two parts. On hardware side, we would like to further enhance the resolution of the iris images by using better IR imaging sensors. The lens module can also be improved in order to better focus on the iris region instead of the eyelids or eyelashes. On software side, we are considering replacing traditional iris segmentation method with deep learning framework. Deep learning has been proved to be a very powerful tool in many computer vision and pattern recognition problems. By applying deep learning, the accuracy of iris segmentation is expected to be enhanced to nearly perfect, and such system will be ready for commercialization.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

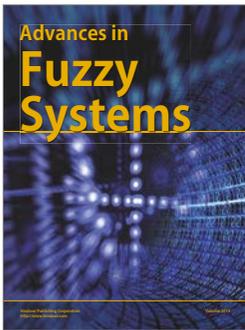
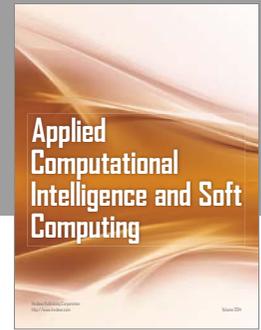
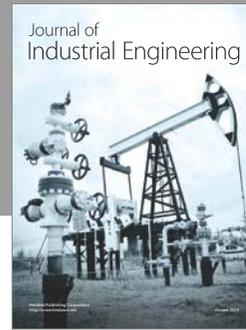
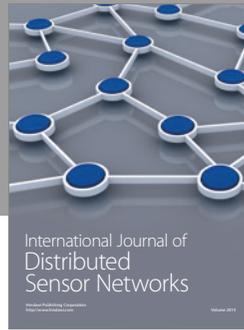
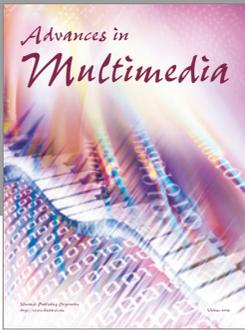
Acknowledgments

This study was accomplished with the support of the National Science Council in Taiwan under Contract no. MOST 105-2221-E-008-111.

References

- [1] The Verge, 2016, "How to fake a fingerprint and break into a phone," <https://www.youtube.com/watch?v=tj2Ty7WkGqk>.
- [2] J. Daugman, "How iris recognition works," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 21–30, 2004.
- [3] J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1148–1161, 1993.
- [4] J. Daugman, "The importance of being random: statistical principles of iris recognition," *Pattern Recognition*, vol. 36, no. 2, pp. 279–291, 2003.
- [5] R. P. Wildes, "Iris recognition: an emerging biometric technology," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1348–1363, 1997.
- [6] J. Daugman, "New methods in iris recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 37, no. 5, pp. 1167–1175, 2007.
- [7] W. W. Boles and B. Boashash, "A human identification technique using images of the iris and wavelet transform," *IEEE Transactions on Signal Processing*, vol. 46, no. 4, pp. 1185–1188, 1998.
- [8] Y. Zhu, T. Tan, and Y. Wang, "Biometric personal identification based on iris patterns," in *Proceedings of 15th International Conference on Pattern Recognition (ICPR '00)*, vol. 2, pp. 801–804, September 2000.
- [9] C. Sanchez-Avila, R. Sanchez-Reillo, and D. De Martin-Roche, "Iris-based biometric recognition using dyadic wavelet transform," *IEEE Aerospace and Electronic Systems Magazine*, vol. 17, no. 10, pp. 3–6, 2002.
- [10] J. S. Arteaga-Falconi, H. Al Osman, and A. El Saddik, "ECG authentication for mobile devices," *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 3, pp. 591–600, 2016.
- [11] S. J. Kang, S. Y. Lee, H. I. Cho, and H. Park, "ECG authentication system design based on signal analysis in mobile and wearable devices," *IEEE Signal Processing Letters*, vol. 23, no. 6, pp. 805–808, 2016.
- [12] T. Hoang, D. Choi, and T. Nguyen, "Gait authentication on mobile phone using biometric cryptosystem and fuzzy commitment scheme," *International Journal of Information Security*, vol. 14, no. 6, pp. 549–560, 2015.
- [13] V. M. Patel, R. Chellappa, D. Chandra, and B. Barbelo, "Continuous user authentication on mobile devices: recent progress and remaining challenges," *IEEE Signal Processing Magazine*, vol. 33, no. 4, pp. 49–61, 2016.
- [14] E. Khoury, L. El Shafey, C. McCool, M. Günther, and S. Marcel, "Bi-modal biometric authentication on mobile phones in challenging conditions," *Image and Vision Computing*, vol. 32, no. 12, pp. 1147–1160, 2014.
- [15] C. Galdi, M. Nappi, and J.-L. Dugelay, "Multimodal authentication on smartphones: combining iris and sensor recognition for a double check of user identity," *Pattern Recognition Letters*, 2015.
- [16] M.-K. Lee, "Security notions and advanced method for human shoulder-surfing resistant PIN-entry," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 4, pp. 695–708, 2014.
- [17] M.-K. Lee, J. B. Kim, and M. K. Franklin, "Enhancing the security of personal identification numbers with three-dimensional displays," *Mobile Information Systems*, vol. 2016, Article ID 8019830, 2016.

- [18] J.-N. Luo and M.-H. Yang, "A mobile authentication system resists to shoulder-surfing attacks," *Multimedia Tools and Applications*, vol. 75, no. 22, pp. 14075–14087, 2015.
- [19] X. Zhao, T. Feng, W. Shi, and I. A. Kakadiaris, "Mobile user authentication using statistical touch dynamics images," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 11, pp. 1780–1789, 2014.
- [20] M. Martinez-Diaz, J. Fierrez, and J. Galbally, "Graphical password-based user authentication with free-form doodles," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 4, pp. 607–614, 2016.
- [21] D. Sliney, D. A. Rosa, F. DeLori et al., "Adjustment of guidelines for exposure of the eye to optical radiation from ocular instruments: statement from a task group of the international commission on non-ionizing radiation protection (ICNIRP)," *Applied Optics*, vol. 44, no. 11, pp. 2162–2176, 2005.
- [22] S. Barra, A. Casanova, F. Narducci, and S. Ricciardi, "Ubiquitous iris recognition by means of mobile devices," *Pattern Recognition Letters*, vol. 57, pp. 66–73, 2015.
- [23] R. Kerekes, B. Narayanaswamy, J. Thornton, M. Savvides, and B. V. K. Vijaya Kumar, "proceedings of the Graphical model approach to iris matching under deformation and occlusion," in *2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, Minneapolis, MN, USA, June 2007.
- [24] J. Thornton, M. Savvides, and B. V. K. V. Kumar, "A Bayesian approach to deformed pattern matching of iris images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 596–606, 2007.
- [25] W. K. Kong and D. Zhang, "Accurate iris segmentation based on novel reflection and eyelash detection model," in *proceedings of the 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing (ISIMP '01)*, pp. 263–266, May 2001.
- [26] Y.-H. Li and M. Savvides, "Automatic iris mask refinement for high performance iris recognition," in *proceedings of the 2009 IEEE Workshop on Computational Intelligence in Biometrics: Theory, Algorithms, and Applications (CIB '09)*, pp. 52–58, April 2009.
- [27] Y.-H. Li and M. Savvides, "A pixel-wise, learning-based approach for occlusion estimation of iris images in polar domain," in *proceedings of the 2009 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '09)*, pp. 1357–1360, April 2009.
- [28] Y.-H. Li and M. Savvides, "Fast and robust probabilistic inference of iris mask," in *SPIE Defense & Security Symposium on Biometric Identification Technologies, Proc. SPIE 7306, 73062I*, 2009.
- [29] Y.-H. Li and M. Savvides, "An automatic iris occlusion estimation method based on high-dimensional density estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 4, pp. 784–796, 2013.
- [30] S. Lim, K. Lee, O. Byeon, and T. Kim, "Efficient iris recognition through improvement of feature vector and classifier," *ETRI Journal*, vol. 23, no. 2, pp. 61–70, 2001.
- [31] M. De Marsico, M. Nappi, D. Riccio, and H. Wechsler, "Mobile Iris Challenge Evaluation (MICHE)-I, biometric iris dataset and protocols," *Pattern Recognition Letters*, vol. 57, pp. 17–23, 2015.



Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

