Multimedia Systems (2013) 19:117–129
DOI 10.1007/s00530-012-0283-z

REGULAR PAPER

Implementation and evaluation of a remote authentication system using touchless palmprint recognition

Haruki Ota · Shoichiro Aoyama · Ryu Watanabe ·
Koichi Ito · Yutaka Miyake · Takafumi Aoki

Published online: 24 July 2012

Abstract When a cellular phone is lost or stolen, it may be used improperly or the personal information may be stolen from it by a malicious user. Biometric authentication such as palmprint recognition is the strongest of the personal authentication technologies designed to prevent such misuse. In biometric authentication, when compared with a local authentication model, a remote authentication model has several advantages such as direct authentication and authentication levels. Ito et al. proposed several palmprint recognition schemes using correspondence matching based on the phase-only correlation. However, these schemes require a palmprint image to be captured with the hand touching the dedicated device, while palmprint images must be captured without such physical contact when using cellular phones. Thus, these schemes cannot be applied to cellular phones since there are large positioning gaps and large differences in brightness and distortion between the images. Furthermore, they have not been implemented in cellular phones and their performances have not been evaluated either. In this paper, we adopt a remote authentication model from the two types of biometric authentication incorporating the above advantages and propose a remote system between a cellular phone and an authentication server. We implement the proposed system using two different types of Android terminal as the terminal on the user side. We also show the validity of the proposed system by examining and confirming the accuracy and processing time. We furthermore discuss the problem of an impersonation attack on the proposed system and consider solutions to this problem from the viewpoints of security and usability. Then, we adopt a palmprint recognition scheme as a biometric authentication scheme and, in particular, use a palmprint recognition algorithm that incorporates Yörük et al.'s preprocessing technique to Ito et al.'s and Iitsuka et al.'s schemes.

Keywords Android · Carried camera · Remote system · Touchless palmprint recognition · Impersonation attack

H. Ota (✉) · R. Watanabe · Y. Miyake
KDDI R&D Laboratories, Inc, 2–1–15 Ohara,
Fujimino-shi, Saitama, Japan
e-mail: haruki@kddilabs.jp

S. Aoyama · K. Ito · T. Aoki
Tohoku University, 6–6–05 Aoba Aramaki, Aoba-ku,
Sendai-shi, Miyagi, Japan

1 Introduction

1.1 Background

A personal authentication technology serves to identify a person, which requires authentication using information registered at an earlier time. This technology can be implemented on these bases: (1) authentication based on personal knowledge (what an individual knows), (2) authentication based on an individual's possessions (what the individual owns) and (3) authentication based on an individual's physical characteristics (what an individual is) or behavioral patterns (what an individual does) [1]. Basis (3) is called biometric authentication (or simply biometrics), which is the main theme of this paper. Biometric authentication can be implemented in various ways: schemes based on physical characteristics such as fingerprint, face, iris and palmprint, and schemes based on behavioral patterns such as voice, signature and keystrokes.

Many people always carry their cellular phones with them. A sophisticated cellular phone, which stores a lot of personal information, can be used for electronic account settlement in online shopping and banking. When a cellular phone is lost or stolen, it may be improperly used or the personal information may be stolen from it by a malicious user. A cellular phone has various security functions to prevent such misuse. Biometric authentication is the strongest protection against impersonation by an unauthorized user, and is stronger than security based on (1) or (2) above. Therefore, the cellular phones require fingerprint, face, voice or signature recognition.

1.2 Biometric authentication models

Biometric authentication models are of two types: local and remote authentication. In the former model, the biometric data are matched on the user side. The fingerprint, face, voice and signature recognitions currently carried in the cellular phone are used in the local authentication model. However, strictly speaking, a user is not authenticated as the genuine user in the local biometric authentication model when he/she wants to receive the authentication service of the cellular phone in a remote environment. This is because only the authentication result is sent to service providers from the cellular phone. In this case, another technology, such as public key infrastructure (PKI), must be used in conjunction with local biometric authentication.

In contrast, the biometric data are matched on the server side in the remote authentication model. In this model, a user can be authenticated directly as the genuine user when he/she wants to receive authentication from the cellular phone in a remote environment. Also, the security level of biometric authentication is provided uniquely through an algorithm and a fixed threshold value. However, there are various services provided for cellular phones in a remote environment. Such services also have various security levels (authentication levels). In this case, service providers want to determine the authentication levels according to the class of service. Therefore, in this paper, we adopt remote authentication as the biometric authentication model.

1.3 Related works

It is generally accepted that among the various methods of biometric authentication, palmprint recognition is more accurate than face, voice and signature recognition, and just as secure as fingerprint recognition [2]. Highly accurate fingerprint recognition and palmprint recognition require the use of a touch-based fingerprint sensor when it is used with a cellular phone. Touchless fingerprint recognition schemes have also been proposed, but they still require dedicated fingerprint sensors (e.g., [3, 4]). To contain cost, we do not want to add a dedicated device to a cellular phone. Furthermore, in these schemes, the fingerprint image can only be captured touchlessly with the finger almost held immobile next to the sensor device. Therefore, these schemes are not suitable for cellular phones. As mentioned above, palmprint recognition is the most promising of the biometric authentication methods used in cellular phones. The palmprint used in this paper is defined in detail in Sect. 2.1.

Baltscheffsky and Anderson [5] were the first to investigate the palmprint as an identity verification target. Kung, Lin and Fang [6] designed a palm recognition system adapted from a face recognition system based on decision-based neural networks. Boles and Chu [7] presented a prototype system for human identification using images of the palm. Zhang's research team, which currently leads the field, has proposed many palmprint recognition schemes and developed techniques for palmprint capturing, preprocessing, feature extraction, coding and matching (e.g., [2, 8–15]). Also, many palmprint recognition schemes using approaches based on classifying texture, lines, appearance and features have been proposed by other research teams (e.g., [16–23]). In particular, Ito et al. and Iitsuka et al. have proposed palmprint recognition schemes using correspondence matching based on phase-only correlation (POC) [24–29].

In these existing schemes, a palmprint image is captured by bringing the hand into contact with a dedicated device installed with a camera. However, palmprint images must be captured touchlessly without such dedicated devices when using the camera in a cellular phone. Thus, using either local or remote authentication, the performance of the palmprint recognition scheme must be evaluated for the authentication service of the cellular phone. For the former requirement, the following problems cannot be dealt with in the aforementioned existing schemes because of the discrepancies between the immediate and reference images that occur in the case of a cellular phone.

- The large positioning gaps, which cause translation, rotation and altered size scale
- The large differences in brightness and distortion

The POC used in Ito et al.'s and Iitsuka et al.'s schemes has the advantage that it is comparatively robust against these problems [24–29]. However these schemes cannot resolve all of these problems completely, and require the addition of appropriate preprocessing techniques. To these schemes, then, we apply Yörük et al.'s preprocessing technique [30], which is used in the hand shape-based recognition scheme. For the latter requirement, the existing palmprint recognition schemes are not implemented in cellular phones. Therefore, their performance is not evaluated for the local authentication model, much less the remote authentication model. Accordingly, there is a strong need to conduct practical investigations using cellular phones.

1.4 Contributions

In this paper, from the two types of biometric authentication, we adopt the remote authentication model due to the advantages such as direct authentication and authentication levels and propose a remote system between a cellular phone and an authentication server. We implement the proposed system using a cellular phone as the terminal on the user side. Two types of Android terminal are used as the cellular phones. We also show its validity by confirming and considering its accuracy and processing time. We furthermore discuss a problem with respect to the security of the proposed system, and consider and compare solutions to this problem from the viewpoints of security and usability. We adopt the palmprint recognition scheme as the biometric authentication scheme and, in particular, use the palmprint recognition algorithm that adds Yörük et al.'s preprocessing technique to Ito et al.'s and Iitsuka et al.'s schemes, as described in Sect. 1.3.

1.5 Organization

The rest of this paper is organized as follows. We introduce the definition of the palmprint and the main scheme proposed by Zhang's research team in Sect. 2. We propose a remote system between a cellular phone and an authentication server in Sect. 3. We describe and discuss the evaluation results of the proposed system in Sect. 4. We describe a problem with respect to the security of the proposed system and consider solutions in Sect. 5. Our conclusions are presented in Sect. 6. We describe the improved algorithms of Ito et al. and Iitsuka et al. in Appendix.

2 Preliminaries

2.1 Palmprint

This subsection defines the palmprint as used in this paper.

Strictly speaking, the palmprint is different from "the lines in the palm." The palmprint is the dermatoglyph, that is, the skin ridge patterns, appearing at fixed locales on the palm. The lines in the palm are the creases that are used for fortune-telling, such as the life line, head line and heart line (see Fig. 1). The word "palmprint" is used by Zhang et al., although the lines in the palm are mainly used in the existing schemes. Therefore, we also refer to the dermatoglyph and creases "palmprints" in this paper.

Note that the creases may change over time, although the dermatoglyphs are as unique and permanent as fingerprints. However, the creases are suitable for biometric authentication in cellular phones as these phones are used constantly.

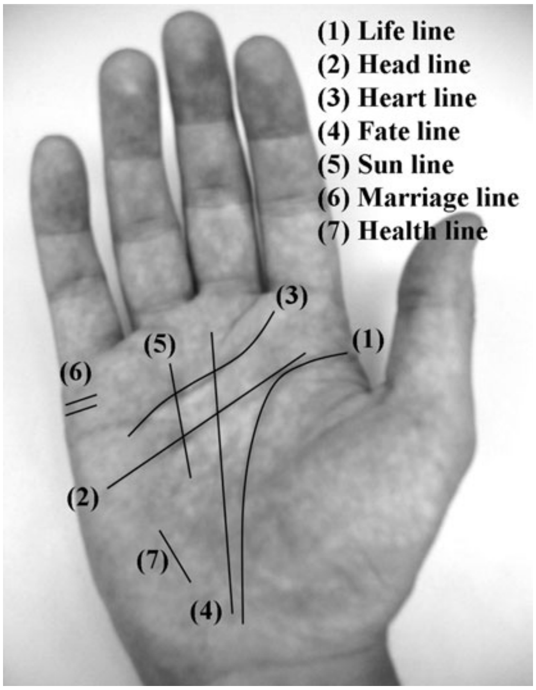


Fig. 1 Creases used for fortune-telling

2.2 Existing scheme

This subsection describes the main scheme proposed by Zhang's research team.

Although Zhang's research team has also proposed many schemes, here we introduce the typical palmprint recognition scheme applicable even to low-resolution palmprint images [10, 11]. The outline of the palmprint recognition scheme proposed by Zhang's research team is as follows.

1. An image of the whole hand with the palm open is captured using the on-device charge-coupled device (CCD) camera. There are marks showing where to place each fingertip.
2. The image captured in step 1 is transformed into a binary image. Then, the boundary of the palm is determined by a boundary tracking algorithm. The coordinates of the central point and an axis are determined from the vertices formed by the junctions of the index and middle fingers and ring and little fingers.
3. A square region of the image is extracted that includes the palmprint, using the central point and axis determined in step 2.
4. An adjusted Gabor filter is applied to the analog information in the square region extracted in step 3, and the information is transformed into a palmprint feature vector.

5. The normalized Hamming distance between the palmprint feature vector from step 4 and the palmprint feature vector generated beforehand in the enrollment procedure is computed as the matching score. Then, the authentication result is judged from this matching score and a fixed threshold value.

In this scheme, a palmprint image is captured using a dedicated device with marks to indicate the placement of each fingertip. Also, the parameters of the adjusted Gabor filter, such as the frequency of the sinusoidal wave, the orientation of the function, and the standard deviation of the Gaussian envelope, must be set up appropriately for the transformation into the palmprint feature vector. This scheme cannot provide superior accuracy when the palmprint recognition scheme is used in a cellular phone. Therefore, another scheme should be required.

3 Proposed system

This section proposes a remote system between a cellular phone and an authentication server.

Figure 2 shows the configuration of the proposed system. The cellular phone and authentication server communicate with each other in a wireless local area network (LAN) environment. The following main functions for the proposed system are implemented in the cellular phone and authentication server, respectively.

- Functions of cellular phone
  1. Capture of palm image: the palm image is captured using the phone's camera.
  2. Transformation of image into data: the captured image is transformed into data using the algorithm described in Appendix.
  3. Adjustment of position: the position of the authenticated image is adjusted to the position of the enrolled data using the algorithm described in Appendix.
  4. Transmission and reception of data: the cellular phone sends data to the authentication server and receives data from it.
  5. Display of information: the information is displayed for the user.
- Functions of authentication server
  1. Enrollment of data in database: the enrolled data and user ID are enrolled in the database.
  2. Search for data in database: the enrolled data are retrieved from the database using the user ID.

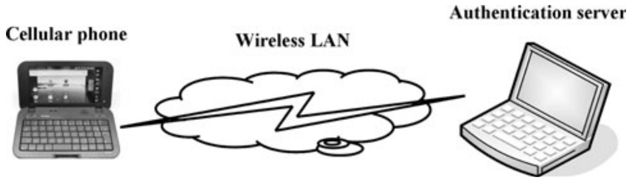


Fig. 2 Configuration of the proposed system

  3. Setup of authentication level: the authentication level is set up for the service class which the user wishes.
  4. Fixing of threshold: the threshold is fixed for the selected authentication level.
  5. Computation of matching score: the matching score between the enrolled and authenticated data is computed using the algorithm described in Appendix.
  6. Judgment of authentication result: the authentication result is judged from the computed matching score and fixed threshold value.
  7. Transmission and reception of data: the authentication server sends data to the cellular phone and receives data from it.

3.1 Enrollment procedure

This subsection explains the enrollment procedure in the proposed system.

We adopt the remote authentication model due to the advantages such as direct authentication and authentication levels, as described in Sect. 1.2. The palmprint data should be stored in the server side since they are matched on it in this model. On the other hand, a user does not want to send a palmprint image to the authentication server without transforming it into other data from the viewpoint of acceptability. Then, we suggest the enrollment procedure in the remote system as follows (see Fig. 3, where the authentication server is abbreviated to "auth. server").

1. A user captures a palm image using the cellular phone camera and stores it together with his/her user ID.
2. The cellular phone transforms the image into the format of the enrolled data using the algorithm.
3. The cellular phone sends the user ID and enrolled data to the authentication server as an enrollment request.
4. The authentication server receives the user ID and enrolled data and enrolls them in the database.
5. The authentication server sends the enrollment result (success or failure) to the cellular phone as the enrollment response.
6. The cellular phone receives the result and displays it for the user.

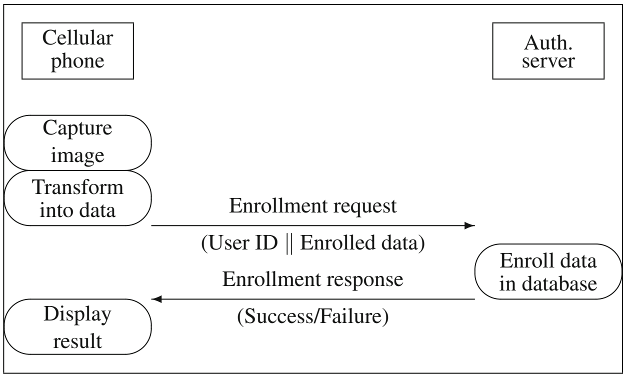


Fig. 3 Enrollment procedure

3.2 Matching procedure

This subsection explains the matching procedure in the proposed system.

In the case of using a cellular phone, there are large positioning gaps and large differences in brightness and distortion between both images, as described in Sect. 1.3. The position of the authenticated image must be adjusted using the enrolled data. On the other hand, users want to use the proposed system not only in their own cellular phones but also in the cellular phones of others from the viewpoint of usability. Therefore, the enrolled data should be sent from the authentication server to the cellular phone. Then, we suggest the matching procedure in the remote system as follows (see Fig. 4, where the authentication level is abbreviated to “auth. level”).

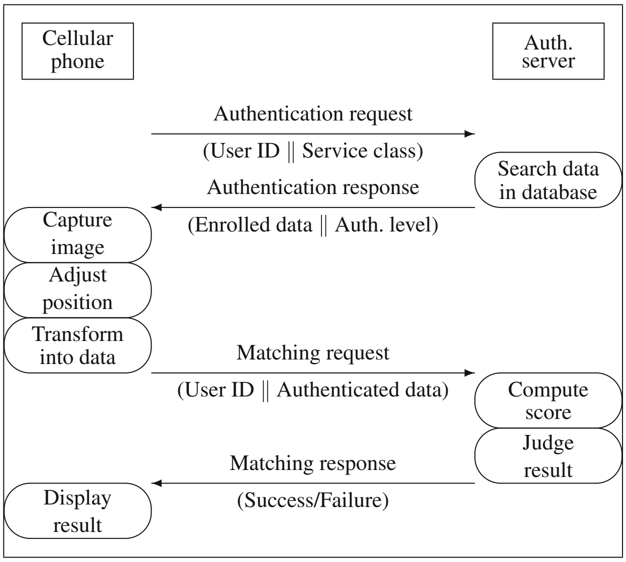


Fig. 4 Matching procedure

1. The cellular phone sends the user ID and service class selected by the user to the authentication server as the authentication request.
2. The authentication server receives the user ID and service class and searches for the enrolled data in the database using the user ID.
3. The authentication server sends the enrolled data and authentication level that corresponds to the service class to the cellular phone as the authentication response.
4. The user captures a palm image using the camera and stores it together with his/her user ID.
5. The cellular phone adjusts the position of the image to that of the enrolled data using the algorithm.
6. The cellular phone transforms the adjusted image into the format of the authenticated data using the algorithm.
7. The cellular phone sends the user ID and authenticated data to the authentication server as a matching request.
8. The authentication server receives the user ID and authenticated data and computes the matching score between both data using the algorithm.
9. The authentication server judges the authentication result from the matching score and the threshold value fixed for the authentication level.
10. The authentication server sends the authentication result (success or failure) to the cellular phone as the matching response.
11. The cellular phone receives the result and displays it for the user.

Remark In the remote authentication model, there are attack scenarios that the local authentication model is not vulnerable to, such as an eavesdropping attack, replay attack and impersonation attack by the server administrator with respect to the biometric data. These attacks can be prevented by cryptosystems, challenge-response mechanisms and template protection techniques, respectively [31]. Note that for the sake of simplicity, discussion of security against the aforementioned attacks is beyond the scope of this paper.

4 Evaluation

This section reports the evaluation of the system proposed in Sect. 3.

4.1 Experimental conditions

This subsection describes the experimental conditions of the proposed system.

Table 1 shows the specifications of the cellular phones and authentication server used in the experiments. We use two types of Android terminal as the cellular phones. Each user captures left-palm images. Figure 5 shows examples of the palmprint images captured by cellular phone 1 (upper) and cellular phone 2 (lower). The following tutorial on palm image capture is shown on the display of the cellular phone (Fig. 6).

- The user takes his/her left palm while holding the cellular phone with his/her right hand.
- The user extends his/her fingers and turns his/her left palm sideways.
- The user adjusts his/her left palm to match the guide mark (see Fig. 6, where the left-hand figure is of cellular phone 1 and the right-hand figure is of cellular phone 2).
- The user selects a background with dark colors and enough light for the photograph.

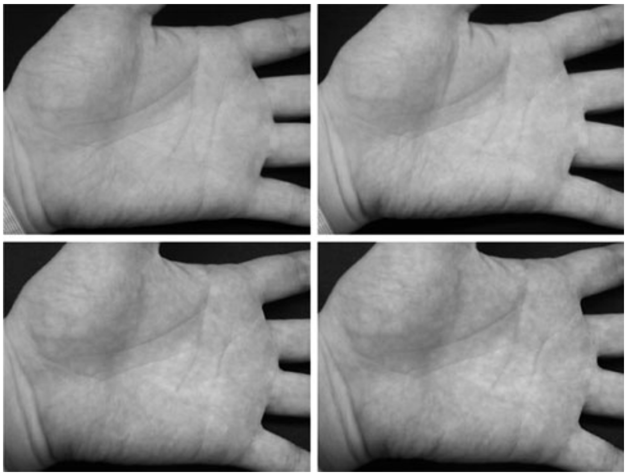
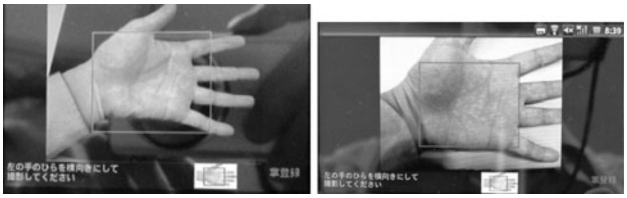


Fig. 5 Examples of palmprint images captured by cellular phone 1 (*upper*) and cellular phone 2 (*lower*)



左の手のひらを横向きにして
撮影してください

左の手のひらを横向きにして
撮影してください

Fig. 6 Guides to capture left-palm image in cellular phone 1 (*left*) and cellular phone 2 (*right*)

4.2 Accuracy

This subsection reports on the accuracy of the proposed system.

4.2.1 Cellular phone 1

When using cellular phone 1 (Android Dev Phone 1/Dev Phone 2), each of 12 users captures 5 left-palm images, for a total of 60 images. There are a total of 1,770 possible combinations for the 60 images. First, we compute the false rejection rate (FRR) for all 120 possible combinations of genuine users. Next, we compute the false acceptance rate (FAR) for all 1,650 possible combinations of impostors. Then, we perform an evaluation using the equal error rate (EER), i.e. the error rate when the FRR and FAR are equal.

Table 1 Specifications of cellular phones and authentication server

Cellular phone 1	
Hardware	Android Dev Phone 1/Dev Phone 2
CPU	Qualcomm MSM7201A 528 MHz
RAM	192 MB
OS	Android 1.6 r1
Camera	3.2-megapixels resolution with autofocus
Cellular phone 2	
Hardware	IS01 kit for developers
CPU	Qualcomm Snapdragon QSD850 1GHz
RAM	192 MB
OS	Android 1.6
Camera	5.27-megapixels resolution with autofocus
Authentication server	
CPU	Intel Core2 Duo processor SU9600 1.6 GHz
RAM	4 GB
OS	CentOS 5.4

Figure 7 shows the FRR and FAR of the proposed system using cellular phone 1, where the horizontal axis indicates the threshold value and the vertical axis indicates the error rates for the FRR and FAR. Note that the FRR is not a smooth curve since there are only 12 users in this experiment. From Fig. 7, the EER is 3.3 % and the corresponding threshold value is 0.263. The FRR is 5.0 % when the FAR is 0 % and the corresponding threshold value is 0.320. That is, the proposed system can also accept genuine users reasonably well even when set to completely reject impostors. On the other hand, the FAR becomes very large when the FRR is 0 %. However, the FAR is 8.8 % when the FRR is 2.8 %, which is 0.5 % smaller than the EER, and the corresponding threshold value is 0.244. That is, the proposed system can considerably reject impostors when it is set to accept many genuine users as genuine. Therefore, in the proposed system, it is necessary to make the minimum threshold value larger than 0.244 and it is permissible to make the maximum threshold value somewhat larger than 0.320 when it sets up various authentication levels.

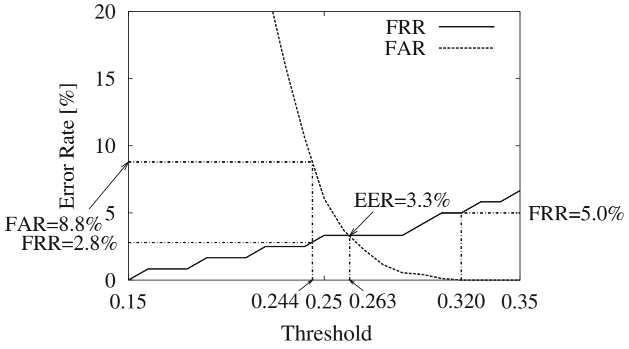


Fig. 7 FRR and FAR of the proposed system using cellular phone 1

4.2.2 Cellular phone 2

When using cellular phone 2 (IS01 kit for developers), each of 8 users captures 5 left-palm images, for a total of 40 images. There are a total of 780 possible combinations for the 40 images. First, we compute the FRR for all 80 possible combinations of genuine users. Next, we compute the FAR for all 700 possible combinations of impostors. Then, we perform an evaluation using the EER.

From Fig. 8, the EER is 2.5 % and the corresponding threshold value is 0.227. The FRR is 5.0 % when the FAR is 0 % and the corresponding threshold value is 0.270. That is, the proposed system can also accept genuine users reasonably well even when set to completely reject impostors. On the other hand, the FAR becomes relatively large when the FRR is 0 %. However, the FAR is 7.4 % when the FRR is 2.0 %, which is 0.5 % smaller than the EER, and the corresponding threshold value is 0.206. That is, the proposed system can considerably reject impostors when it is set to accept many genuine users as genuine. Therefore, in the proposed system, it is necessary to make the minimum threshold value larger than 0.206 and it is permissible to make the maximum threshold value somewhat larger than 0.270 when it sets up various authentication levels.

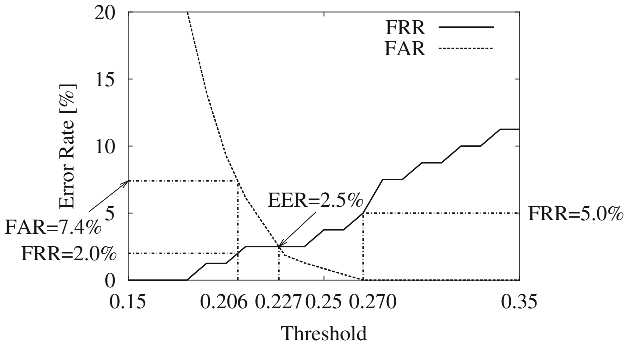


Fig. 8 FRR and FAR of the proposed system using cellular phone 2

4.3 Discussion

This subsection considers the validity of the proposed system.

First, we compare the proposed system with the following existing schemes before considering its validity.

- The typical scheme developed by Zhang's research team [11] described in Sect. 2.2.
- The newest scheme developed by Ito et al. [29], which is the source of the algorithm described in Appendix.

We show the EER of each scheme/system in Table 2, where the values of the existing schemes are those presented in [29]. Note that the experimental conditions are different in our system and the existing schemes. From Table 2, Ito et al.'s scheme has higher accuracy than Zhang et al.'s scheme, and our system has the lowest accuracy of these schemes and systems. However, in these two schemes, the palmprint image is captured by bringing the hand into contact with a dedicated device in accordance with fingertip placement marks. In this case, the positions, directions and sizes of the enrolled and authenticated palm images become almost the same. On the other hand, the palmprint image is captured touchlessly using the cellular phone camera in the proposed system. As described above, there are large positioning gaps and large differences in brightness and distortion between both images. Therefore, it is natural that our system has lower accuracy than the two existing schemes.

However, the proposed system is based on Ito et al.'s scheme plus Yörük et al.'s preprocessing technique [30]. The POC used in Ito et al.'s scheme is relatively robust when dealing with the problems posed by the positioning gaps and differences in brightness and distortion between the enrolled and authenticated images in the cellular phone. Therefore, Ito et al.'s scheme is superior to Zhang et al.'s scheme with respect to both accuracy and applying it to a cellular phone.

Consequently, our system can be expected to have higher accuracy than Zhang et al.'s scheme when our system and their scheme use the palmprint images captured touchlessly by a cellular phone.

Table 2 Comparisons of the proposed system and existing schemes

System/schemes		EER (%)	Computation time
Zhang et al. [11]		0.3919	1.2391
Ito et al. [29]		0.0000	0.9216
Our system	Phone 1	3.3	See Table 3
	Phone 2	2.5	See Table 4

4.4 Processing time

This subsection evaluates the processing time of the proposed system.

We show the computation time of each scheme/system in Table 2, where the values of the existing schemes are those presented in [29]. Tables 3 and 4 show the processing time (computation and communication time) of the proposed system using cellular phones 1 and 2, respectively. In Table 4, the computation time of the preprocessing stage is included in the communication time of the matching request and response phase since it is very short.

From Table 3, the computation time of the proposed system for cellular phone 1 is about 0.94 s, where the computation time of the preprocessing stage is about 0.8 s in the cellular phone and the computation time of the matching stage is about 0.14 s in the authentication server. Also, the communication time of the proposed system for cellular phone 1 is about 5.48 s. In this case, the authentication request and response phase take 2.37 s including the search step for enrolled data in the database, and the matching request and response phase take 3.11 s. The total processing time of the proposed system for cellular phone 1 is 6.42 s.

From Table 4, the computation time of the proposed system for cellular phone 2 is about 0.14 s. As described above, the computation time of the preprocessing stage is very short. Consequently, this time corresponds to that of the matching stage in the authentication server. Also, the communication time of the proposed system for cellular phone 2 is about 2.52 s. In this case, the authentication request and response phase take 1.71 s including the search step for enrolled data in the database, and the matching request and response phase take 0.81 s. The total processing time of the proposed system for cellular phone 2 is 2.66 s.

These results show that the proposed system has a practical processing time.

Table 3 Computation time (upper) and communication time (lower) of the proposed system for cellular phone 1 s

Preprocessing stage	Matching stage	Subtotal
0.8	0.14	0.94
Authentication request/response	Matching request/response	Subtotal
2.37	3.11	5.48

Table 4 Computation time (upper) and communication time (lower) of the proposed system for cellular phone 2 s

Preprocessing stage	Matching stage	Subtotal
Very short	0.14	0.14
Authentication request/response	Matching request/response	Subtotal
1.71	0.81	2.52

5 Security

This section considers the security of the system proposed in Sect. 3.

5.1 Problem

This subsection discusses a security-related problem of the proposed system.

Palmprint images must be captured touchlessly without a dedicated device when using the camera in a cellular phone, as described in Sect. 1.3. As a result, there are large positioning gaps and large differences in brightness and distortion between both the immediate and reference images. Matching between both images whose positions are not adjusted cannot be expected to have superior accuracy. In the palmprint recognition algorithm used in our system, the position of the authenticated image is adjusted using the enrolled data. Thus, the flow of sending the enrolled data from the authentication server to the cellular phone is included in step 3 of the matching procedure of the proposed system.

However, in the following attack scenario (Fig. 9), an attacker can impersonate the legitimate user by sending the authentication server the enrolled data that were sent from it as the authenticated data. Therefore, we assume that the attacker can obtain the user ID beforehand or use it freely.

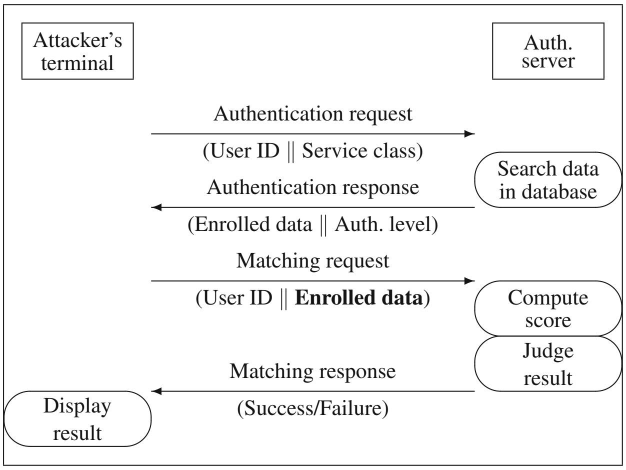


Fig. 9 Attack scenario

1. The attacker's terminal sends the user ID and service class selected by the user to the authentication server as the authentication request.
2. The authentication server receives the user ID and service class and searches for the enrolled data in the database using the user ID.
3. The authentication server sends the enrolled data and authentication level that corresponds to the service class to the attacker's terminal as the authentication response.
4. The attacker's terminal sends the user ID and enrolled data instead of the authenticated data (hereinafter shown as "**Enrolled data**" in bold-faced type) to the authentication server as a matching request.
5. The authentication server receives the user ID and **Enrolled data** and computes the matching score between both data using the algorithm.
6. The authentication server judges the authentication result from the matching score and the threshold value fixed for the authentication level.
7. The authentication server sends the authentication result (success or failure) to the attacker's terminal as the matching response.
8. The attacker's terminal receives the result and displays it for him/her.

In the above scenario, the attacker sends the same data as those enrolled in the database of the authentication server to the server, instead of the authenticated data. In this case, the attacker succeeds in impersonating the legitimate user since the matching score between both data becomes 1 (or close to 1).

5.2 Solutions

This subsection considers the solutions to the problem discussed in the previous subsection.

There are three solutions to the problem described in Sect. 5.1 as follows:

(I) Another threshold is added for the matching score between the enrolled and authenticated data.
(II) The authentication server ceases sending the enrolled data to the cellular phone.
(III) The enrolled data sent from the authentication server to the cellular phone are transformed so that the attacker cannot misuse them.

5.2.1 Solution (I)

Generally, the enrolled and authenticated data rarely become completely identical since noise occurs in the image capturing step of biometric authentication. If both data are completely identical (or almost identical), there is a method that treats this as an aberration and the authentication result is judged to be a failure. An impersonation attack on the proposed system can be also prevented by the above method. However, we can assume that the attacker will alter the enrolled data and thereby lower the matching score so that it falls within the range in which the authentication result is judged to be a success. Therefore, this solution is not sufficient from the viewpoint of security.

5.2.2 Solution (II)

The simplest solution is to cease sending the enrolled data from the authentication server to the cellular phone since sending this data is the essential problem. It is necessary for the enrolled data to be stored in the cellular phone in order to adjust the position of the authenticated image using the enrolled data. To ensure security against an impersonation attack, the enrolled data must be stored in a tamper-resistant device such as an IC card to prevention leakage.

However, in this paper, we assume the existence of personal authentication services in the remote environment. In this case, it is preferable for the users to be able to receive such services not only in their own cellular phones but also in the cellular phones of others. However, the user cannot receive the services in someone else's cellular phone when the enrolled data must be stored in his/her cellular phone. This is the reason why the flow of sending the enrolled data is included in the proposed system. The enrolled data may be stored in other media that can be used in the cellular phone such as a micro secure digital (SD) card. However, it is troublesome for the users to carry the media with them and use it as this involves additional actions. Therefore, this solution is unacceptable from a usability viewpoint.

5.2.3 Solution (III)

We want to resolve the problem with respect to an impersonation attack while sending the enrolled data from the authentication server to the cellular phone without adding another threshold for the matching score, as is needed in solutions (I) and (II). Then, the enrolled data must be transformed so that the attacker cannot misuse the data. That is, we consider a method of dividing the palmprint image into two types of blocks: blocks to adjust the position of the authenticated image and blocks to match the enrolled and authenticated data. This is because it is extremely difficult for the attacker to impersonate the legitimate user using the blocks for matching even if he/she misuses only the blocks for adjustment. Figure 10 shows an example of the two types of blocks, those for adjustment (ADJ) and those for matching (MAT) in the palmprint image. The palmprint image is divided into 25 blocks of 5 × 5 in this example. The center block and four blocks adjacent to this block in the diagonal directions are used for matching among these blocks. The 12 blocks adjacent to these 5 blocks in the vertical and horizontal directions are used for adjustment. The remaining eight blocks are not used. The size of the palmprint image is 160 × 160 pixels. The average translation of the block for matching becomes the average translation of the four neighboring blocks for adjustment. The enrollment and matching procedures in the proposed system are revised as follows, where we show only the steps modified from the initial system.

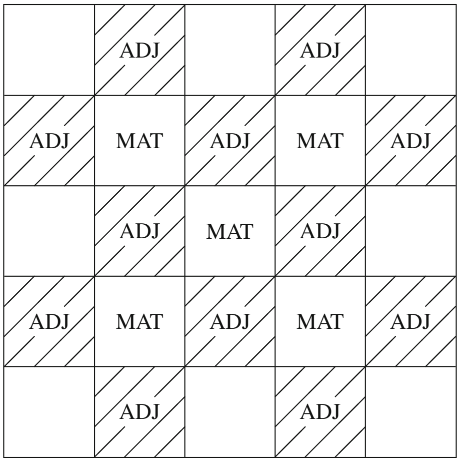
- Enrollment procedure

  2. The cellular phone transforms all the 25-block images into the format of the enrolled data using the algorithm.

- Matching procedure

  3. The authentication server sends the 12-block enrolled data for adjustment and the authentication level that corresponds to the service class to the cellular phone as the authentication response.
  6. The cellular phone transforms the 5-block adjusted image for matching into the format of the authenticated data using the algorithm.
  8. The authentication server receives the user ID and authenticated data and computes the matching score between both the 5-block data for matching using the algorithm.

However, in this solution, the performance when adjusting the position of the authenticated image may be degraded compared with that in the initial system. This is because only 12 blocks are used for adjustment in the former whereas all 25 blocks are used in the latter. We conduct an experiment with respect to solution (III) using the PolyU Palmprint Database Version 1 [32] in order to confirm the performance when adjusting the position[1]. Each of 100 users captures 6 left-palm images, for a total of 600 images. There are a total of 1,500 possible combinations of genuine users and a total of 178,200 possible combinations of impostors.



	ADJ		ADJ	
ADJ	MAT	ADJ	MAT	ADJ
	ADJ	MAT	ADJ	
ADJ	MAT	ADJ	MAT	ADJ
	ADJ		ADJ	

Fig. 10 An example of blocks for adjustment (ADJ) and blocks for matching (MAT) in palmprint image

Figure 11 shows the receiver operating characteristic (ROC) curve for the experiment with respect to solution (III). The ROC curve denotes the plots of the FAR for the FRR at different threshold values on the matching score. In the left-hand graph, “Initial” denotes the initial system, and “Prop A” and “Prop B” denote the proposed schemes A and B using solution (III). In the right-hand graph, “Zhang” denotes Zhang et al.’s scheme [11], which is examined under the same conditions. In the initial system and proposed scheme A, the highest peak value of the average band-limited POC (BLPOC) function between each block is computed as the matching score. In proposed scheme B, the highest peak value in the central 5 × 5 pixels of the BLPOC function between each block is computed, and their average value is computed as the matching score. Then, we evaluate the accuracy of these systems and schemes using the EER. From the left-hand graph of Fig. 11, the EERs of the proposed schemes A and B are 0.322 and 0.069 %, respectively, and that of the initial system is 0.008 %. As shown in the right-hand graph of Fig. 11, the EER of Zhang et al.’s scheme is 2.147 % [27]. Therefore, the accuracy of solution (III) is sufficiently high although it is not as good as that of the initial system.

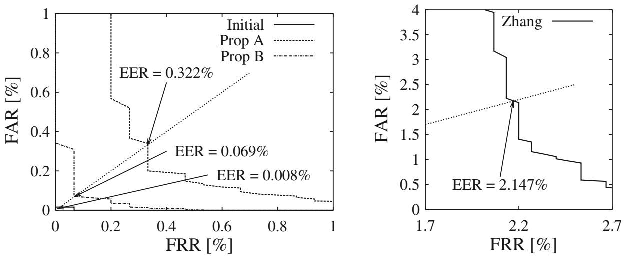


Fig. 11 ROC curve for experiment with respect to solution (III)

[1] The aim of this experiment is to confirm the performance when adjusting the position. We use the palmprint images captured with the hand touching the dedicated device, which has smaller positioning gaps. For these images, we use the preprocessing technique of Zhang’s research team [11] to obtain the palmprint region.

5.3 Comparison

This subsecion compares the initial system and the proposed solutions.

Table 5 shows a comparison of the initial system and the solutions, where "Y" denotes that each system/solution satisfies each criterion, "y" denotes that it is probable that the criterion is satisfied, and "N" denotes that the criterion is not met. The initial system can be applied to the cellular phones of others and is expected to be highly accurate. However, it is vulnerable to an impersonation attack, as shown in Sect. 5.1. Solution (I) can prevent an impersonation attack on the initial system but is not secure against a modified impersonation attack. Solution (II) cannot be applied to the cellular phones of others, however, it is secure against the above two attacks. Solution (III) is secure against impersonation attacks and can also be applied to the cellular phones of others. In solution (III), accuracy is also degrated due to the performance degradation for position adjustment. However, it is difficult to equal both the usability and security of a remote system when applying solutions (I) and (II). On the other hand, solution (III) can equal both the usability and security of a remote system by alleviating the above performance degradation. Therefore, solution (III) is the most promising of these three solutions.

5.4 Conclusions

In this paper, we explained that in biometric authentication, when compared with local authentication, a remote authentication model has several advantages such as direct authentication and authentication levels. We adopted remote authentication as the biometric authentication model due to these advantages and proposed a remote system between a cellular phone and an authentication server. We implemented the proposed system on two types of Android terminal as the terminal on the user side, and conducted experiments on accuracy and processing time. We confirmed that the EERs of the proposed system are 3.3 and 2.5 % for the two types of Android terminal, respectively, and showed the validity of our system indirectly by comparing it with existing schemes and examining these schemes. We confirmed that the total processing time of the proposed system is 6.42 and 2.66 s for them, respectively, which represent acceptable times in practice. We discussed the problem of an impersonation attack on the proposed system and proposed the three solutions to this problem. We confirmed that the solution of dividing the palmprint image into blocks for adjustment and matching is the most promising of the three solutions by considering and comparing them from the viewpoints of the security and usability.

Table 5 Comparison of initial system and solutions

System/solutions	Usability	Security	
	Others' terminal	Attack tolerance	Accuracy
Initial system	Y	N	Y
Solution (I)	Y	N	Y
Solution (II)	N	Y	Y
Solution (III)	Y	Y	y

Appendix: Palmprint recognition algorithm

This appendix describes the palmprint recognition algorithm used in this paper.

This palmprint recognition algorithm consists of a preprocessing stage of six steps including Yörük et al.'s technique and a matching stage of two steps.

Preprocessing stage

This subsection describes the six-step preprocessing stage in the palmprint recognition algorithm.

The preprocessing stage consists of the following six steps, where the fifth step includes Yörük et al.'s preprocessing technique.

1. Extraction of an image.
2. Reduction of the extracted image.
3. Flesh color detection based on HSV color system.
4. Opening.
5. Detection of key points.
6. Extraction of palmprint region.

Extraction of image

The right half of an input image is extracted, because only the right half of the input image is required to detect the key points for the extraction of the palmprint region. It is possible to shorten the processing time with this extraction. In this algorithm, the size of the captured image is 1,280 × 960 pixels, the size of the input image that is downsampled is 640 × 480 pixels and the size of the extracted image is 320 × 480 pixels. The guide for image capture is displayed when a user captures an image of his/her hand. At this time, background colors are colors other than flesh color and its related colors, such as red, orange and yellow.

Reduction of extracted image

The image is reduced from the extracted image to half size, which can further shorten the processing time. In this algorithm, the size of the reduced image is 160 × 240 pixels.

Flesh color detection based on HSV color system

The reduced image is initially represented in the RGB color system, then converted to the HSV color system, enabling robust detection of flesh color for a conversion of a bright value. The palm can then be detected by its flesh color. In this algorithm, the color of some image blocks is judged to be flesh color when the H channel of these blocks satisfies the following conditions:

$$\begin{cases} 0 \le H \le 50 \\ 300 \le H \le 360 \end{cases}$$

The image is converted into a binary image of the flesh color domain and the other color domain using the H channel.

Opening

Flesh color may be included in domains other than the palm when that color is detected. These domains are represented as the small connected components. Then, the small connected components other than the palm are eliminated by the opening processing. The opening processing consists of erosion processing and dilation processing. Erosion and dilation are fundamental morphological operations. Erosion is a process to compute the minimum value of pixels inside the kernel region, and to replace the target pixel with this value. Dilation is a process to compute the maximum value of pixels inside the kernel region, and to replace the target pixel with this value. Opening processing involves carrying out the erosion process several times, and then repeating the dilation process the same number of times. In this algorithm, erosion and dilation are each carried out once, and a disk-type structuring element with a radius of 3 pixels is used as the kernel.

Detection of key points

The vertices formed by the junctions of the index and middle fingers, middle and ring fingers and ring and little fingers are detected in order to extract the palmprint region. Figure 12 shows an example of palmprint region extraction.

1. A chain code is generated for the above-mentioned binary image by determining the center of the left end of this image as the starting point (the square point of Fig. 12). The coordinates of the boundaries of the palm can be obtained by generating the chain code.
2. The Euclidean distances between the starting point and coordinates of the boundaries are computed, and illustrated as a graph. In this case, impulse noises are eliminated by a median filter. The valleys in the graph are detected using the slopes of the line passing through the key points. It is possible to detect the vertices between the fingers by detecting the valleys in the graph, since these valleys terminate at the vertices between the fingers. In this algorithm, the vertices between the index and middle fingers and ring and little fingers are determined as the key points (the circle points of Fig. 12).

Extraction of palmprint region

The perpendicular bisectors of the segments connecting the datum points are computed. A point that has some fixed distance from their intersection point is determined to be the centroid (the cross of Fig. 12). A rectangular region image is extracted by setting the centroid as the center of the palmprint region (the square frame of Fig. 12). The extracted image is normalized into an image with 160 × 160 pixels, and is converted into a grayscale image. The RGB image is converted into a YIQ color system image, and the Y channel determines the grayscale image. The grayscale image is called the palmprint region. It is possible to normalize the rotation, expansion and reduction and translation between the palmprint regions to some extent by determining the centroid using the key points.

Matching stage

This subsection describes the two-step matching stage in the palmprint recognition algorithm.

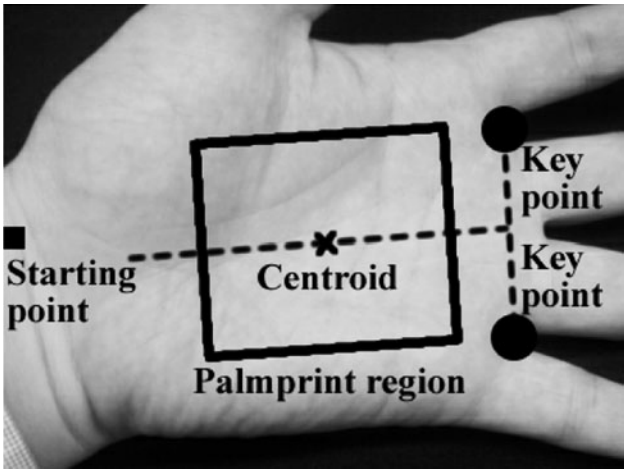


Fig. 12 Example of palmprint region extraction

The matching stage consists of the following two steps:

1. Mapping between the images.
2. Computation of the matching score.

Mapping between images

There are nonlinear distortion and projective transformation between the palm images. These can be approximated by local translation. It is possible to deal with nonlinear distortion and projective transformation by using correspondence matching based on the POC. Also, it is possible to deal with the large positioning gaps, which cannot be normalized in the preprocessing stage. In this algorithm, the block size is 32 × 32 pixels and the number of corresponding points to be found is 16.

Computation of matching score

First, the block image with 32 × 32 pixels is extracted by setting the datum point and corresponding points at the center of this block. The normalized cross power spectrum between corresponding local blocks is computed with the inherent frequency band of palmprint images. Next, the average of all the normalized cross power spectrums is computed. Then, the average BLPOC function is computed by inverse 2-dimensional discrete Fourier transform of the average normalized cross power spectrum. Finally, the highest peak value of the average BLPOC function is computed as the matching score.

References

1. Jain, A., Bolle, R., Pankanti, S.: In: Jain, A., Bolle, R., Pankanti, S. (eds.) BIOMETRICS: Personal Identification in Networked Society, pp. 1–41. Kluwer Academic Publishers, New York (2002)
2. Zhang, D.D.: Palmprint Authentication. Kluwer Academic Publishers, Massachusetts (2004)
3. Song, Y., Lee, C., Kim, J.: In: Proceedings of 2004 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2004), pp. 524–527. IEEE Computer Society (2004)
4. Lee, C., Lee, S., Kim, J.: In: Proceedings of Joint IAPR International Workshop on Structural, Syntactic, and Statistical Pattern Recognition (SSPR2006 and SPR2006). LNCS, vol. 4109, pp. 358–365. Springer, New York (2006)
5. Baltscheffsky, P., Anderson, P.: In: Proceedings of 1986 International Carnahan Conference on Security Technology (ICCST 1986), pp. 229–234. IEEE Computer Society (1986)
6. Kung, S.Y., Lin, S.H., Fang, M.: In: Proceedings of 1995 IEEE Workshop on Neural Networks for Signal Processing, pp. 323–332. IEEE Computer Society (1995)
7. Boles, W., Chu, S.: In: Proceedings of IEEE Region 10 Annual Conference. Speech and Image Technologies for Computing and Telecommunications (TENCON'97) , pp. 295–298. IEEE Computer Society (1997)
8. Shu, W., Zhang, D.D.: In: Proceedings of Digital Image Computing: Techniques and Applications (DICTA 1997), pp. 551–554. Australian Pattern Recognition Society (1997)
9. Shu, W., Zhang, D.D.: Opt. Eng. **37**(8), 2359 (1998)
10. Kong, W.K., Zhang, D.D.: In: Proceedings of 16th International Conference on Pattern Recognition (ICPR 2002), pp. 807–810. IEEE Computer Society (2002)
11. Zhang, D.D., Kong, W.K., You, J., Wong, M.: IEEE Trans. Pattern Anal. Mach. Intell. **25**(9), 1041 (2003)
12. Kong, A.W.K., Zhang, D.D.: In: Proceedings of 17th International Conference on Pattern Recognition (ICPR 2004), pp. 520–523. IEEE Computer Society (2004)
13. Kong, A., Zhang, D.D., Kamel, M.: Pattern Recognit. **39**(3), 478 (2006)
14. Jia, W., Huang, D.S., Zhang, D.D.: Pattern Recognit. **41**(5), 1504 (2008)
15. Zhang, D.D., Guo, Z., Lu, G., Zhang, L., Zuo, W.: IEEE Trans. Instrum. Meas. **59**(2), 480 (2010)
16. Duta, N., Jain, A.K., Mardia, K.V.: Pattern Recognit. Lett. **23**(4), 477 (2002)
17. Han, C.C., Cheng, H.L., Lin, C.L., Fan, K.C.: Pattern Recognit. **36**(2), 371 (2003)
18. Connie, T., Jin, A.T.B., Ong, M.G.K., Ling, D.N.C.: Image Vis. Comput. **23**(5), 501 (2005)
19. Sun, Z., Tan, T., Wang, Y., Li, S.Z.: In: Proceedings of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), pp. 279–284. IEEE Computer Society (2005)
20. Ribaric, S., Fratric, I.: IEEE Trans. Pattern Anal. Mach. Intell. **27**(11), 1698 (2005)
21. Lin, C.L., Chuang, T.C., Fan, K.C.: Pattern Recognit. **38**(12), 2639 (2005)
22. Hu, D., Feng, G., Zhou, Z.: Pattern Recognit. **40**(1), 339 (2007)
23. Hennings-Yeomans, P.H., Kumar, B.V.K.V., Savvides, M.: IEEE Trans. Inf. Forensics Secur. **2**(3), 613 (2007)
24. Ito, K., Aoki, T., Nakajima, H., Kobayashi, K., Higuchi, T.: In: Proceedings of 13th IEEE International Conference on Image Processing (ICIP 2006), pp. 2669–2672. IEEE Computer Society (2006)
25. Ito, K., Aoki, T., Nakajima, H., Kobayashi, K., Higuchi, T.: In: Proceedings of 2006 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2006), pp. 215–218. IEEE Computer Society (2006)
26. Ito, K., Aoki, T., Nakajima, H., Kobayashi, K., Higuchi, T.: IEICE transactions on fundamentals of electronics. Commun. Comput. Sci. **E91-A**(4), 1023 (2008)
27. Iitsuka, S., Ito, K., Aoki, T.: In: Proceedings of 19th International Conference on Pattern Recognition (ICPR 2008), pp. 1–4. IEEE Computer Society (2008)
28. Iitsuka, S., Miyazawa, K., Aoki, T.: In: Proceedings of 16th IEEE International Conference on Image Processing (ICIP 2009), pp. 1973–1976. IEEE Computer Society (2009)
29. Ito, K., Iitsuka, S., Aoki, T.: In: Proceedings of 16th IEEE International Conference on Image Processing (ICIP 2009), pp. 1977–1980. IEEE Computer Society (2009)
30. Yörük, E., Konukoğlu, E., Sankur, B., Darbon, J.: IEEE Trans. Image Process. **15**(7), 1803 (2006)
31. Ota, H., Kiyomoto, S., Tanaka, T.: IEICE transactions on fundamentals of electronics. Commun. Comput. Sci. **E88-A**(1), 287 (2005)
32. The Hong Kong Polytechnic University (PolyU) Palmprint Database. http://www4.comp.polyu.edu.hk/~biometrics/