Three Measures for Secure Palmprint Identification

[1,3]**Adams Kong**, [2]**David Zhang** and [1]**Mohamed Kamel**

[1]Pattern Analysis and Machine Intelligence Lab
Department of Electrical and Computer Engineering
University of Waterloo
200 University Avenue West, Ontario, N2L 3G1, Canada

[2]Biometrics Research Centre
Department of Computing, The Hong Kong Polytechnic University,
Kowloon, Hong Kong

[3]School of Computer Engineering, Nanyang Technological University,
Block N4, Nanyang Avenue, Singapore, 639798

Corresponding author:

Adams Kong

School of Computer Engineering,

Nanyang Technological University,

Block N4, Nanyang Avenue,

Singapore, 639798

Fax: (65) 6792 6559

E-mail: adamskong@ieee.org

Abstract — Most previous research in the area of personal authentication using the palmprint as a biometric trait has concentrated on enhancing accuracy yet resistance to attacks is also a centrally important feature of any biometric security system. In this paper, we address three relevant security issues: template re-issuances, also called [1]cancellable biometrics, replay attacks, and database attacks. We propose to use a random orientation filter bank (ROFB) as a feature extractor to generate noise-like feature codes, called Competitive Codes for templates re-issuances. Secret messages are hidden in templates to prevent replay and database attacks. This technique can be regarded as template watermarking. A series of analyses is provided to evaluate the security levels of the measures.

Keywords: Biometric security, cancellable biometrics, template protection, replay attacks, database attacks.

1. INTRODUCTION

Researchers have proposed various preprocessing, feature extraction, matching, and classification algorithms for on-line palmprint verification and identification [5-8, 10-18, 32]. Most of these have concentrated on improving accuracy by developing new feature extraction and matching algorithms [5-8, 11-18, 32] and by combining palmprint and hand geometric features [5, 13]. Others have proposed hierarchical and classification algorithms [10-12, 14] to alleviate the computational cost of large database identification. Although accuracy and matching speed are important, security of

[1] In this paper, we use the definition of cancelable biometrics proposed by Ratha and his coworkers, who are the first inventors of cancelable biometrics based on our best knowledge [19].

palmprint systems must not to be ignored. Ratha et al draw our attention to the basic vulnerabilities of biometric systems [19-20]. Fig. 1 illustrates a typical biometric system and eight vulnerable points, Points 1-8 [19-20]. The sensor collects biometric signals such as fingerprint images and transmits the signals to the feature extractor through the data link at Point 2. The feature extractor extracts features such as minutiae points in fingerprint images from the biometric signals and transmits them to the matcher through the data link at Point 4. The matcher compares the features and templates from the database to compute a matching score or obtain a decision. Finally, the matching score or the decision is sent to the application.

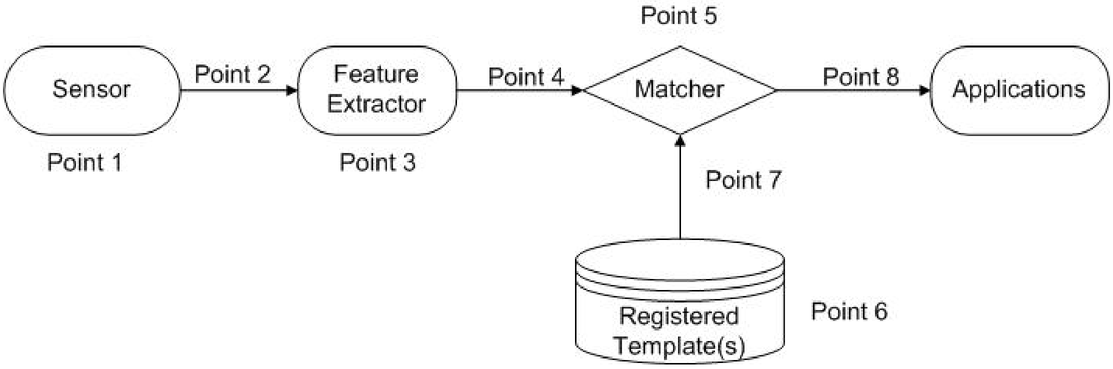

Fig. 1 Vulnerable points in a typical biometric system.

The potential attack points are between and on the common components of a biometric system. These points are especially open to attack when biometric systems are applied to remote, unattended applications, giving attackers enough time to make complex and numerous attempts to break in. At Point 1, fake biometrics such as artificial gummy fingerprints can be used to spoof the systems. At Point 2, it is possible to avoid liveness detection in the sensors by submitting a pre-recorded signal such as a face image. This is a so-called replay attack. At Point 3, it is possible to replace original features with a predefined feature by using a Trojan horse to override the feature extractor. At Point 4, it is possible to apply both brute-force and replay attacks, submitting on the one hand templates stolen from the database or the data link between feature extractor and matcher or, on the other hand, numerous synthetic templates. At Point 5, it is possible to use a Trojan horse to replace original matching scores with preselected matching scores. At Point 6, it is possible to modify templates in the database or insert templates from unauthorized users into the database. At Point 7, replay attacks are once again possible. At Point 8, decision outputs of the system can be overridden and the matching scores can be collected to generate the images in the registered database [9]. In addition to these attacks, Ratha and his co-workers also point out that we have limited biometric traits for template re-issuances.

Although palmprint recognition has been studied for many years and the potential attacks have been identified, only limited research has been done into the security of palmprint systems. A group of researchers developed cancellable palmprints [21-22, 31] for template re-issuance and claimed that their approaches could achieve zero equal error rates. However, their results were based on the unrealistic assumption that users never share or lose their token keys. A detailed analysis is given in [23]. In fact, different features require different cancellable transforms [26, 30]. We cannot directly apply one cancellable transform from one feature to another feature. For analyzing brute-force attacks at Point 4, a probability model has been developed [28]. Nevertheless, the replay attack at Point 4 and the database attack at Point 6 have not been addressed. Although some security measures have been proposed for other biometrics such as fingerprints, they are not especially designed for high-speed palmprint identification [24-25].

Cryptography is one possible solution that would allow us to better defend against replay and database attacks. Systems protected by cryptography store and transmit only encrypted templates in databases and through data links. However, cryptography is not suitable for speed-demanding matching, e.g. real-time large-scale identification, since decryption is required before matching. Another potential solution is cancellable biometrics. Cancellable biometrics transform original templates into other domains and match in the transformed domain. Although cancellable biometrics overcome the weakness of cryptography, current cancellable biometrics are still not secure enough for our palmprint identification. For example, attackers can still insert stolen templates at Point 4 and Point 6 for replay and database attacks before systems can cancel the stolen templates and reissue new templates. Furthermore, current cancellable biometrics cannot detect replay and database attacks. In other words, if attackers insert unregistered templates into data links or databases, systems cannot discover the unregistered templates. To solve these problems, we take advantage of cryptography and cancellable biometrics to design a set of security measures to prevent replay and database attacks for secure palmprint identification.

The rest of this paper is organized as the follows. Section 2 gives a briefly summary of our palmprint identification system based on Competitive Code [18]. Section 3 presents security measures including cancellable Competitive Code, one time pad, and template watermarking. Section 4 reports the trade-off between accuracy and security. Section 5 evaluates the security level of the proposed measures. Section 6 points out some open problems in biometric security. Section 7 offers some concluding remarks.

2. PALMPRINT IDENTIFICATION BASED ON COMPETITIVE CODE

In this section, we provide a brief summary of our palmprint identification system based on [2]Competitive Code [18]. We choose to develop security measures for Competitive Code rather than other palmprint algorithms [6, 8, 11-12, 14, 16-17] since it is the most accurate and the most computationally effective algorithm developed by Zhang and his coworkers.

A palmprint recognition system based on Competitive Code consists of four components: an image acquirer, a preprocessor, a feature extractor, and a matcher, described as follows.

1) The image acquirer: This component collects a palmprint image from the CCD-based palmprint scanner and transmits it to a processor. Fig. 2(a) shows a palmprint scanner [6].

2) The preprocessor: This component aligns different palmprint images based on a coordinate system established by the two key points between fingers [6]. Then, the central parts of palmprint images are extracted according to the coordinate system for feature extraction. Fig. 2(b) and (c) respectively illustrate the coordinate system and a preprocessed palmprint image.

[2] In this paper, Competitive Code interchangeably refers to the method and features presented in [18].

3) The feature extractor: This component employs six real Gabor filters with different orientations to estimate the local orientation field in a preprocessed image *I(x,y)* as features. The real Gabor filters are defined as

$$\psi(x,y,x_o,y_o,\omega,\theta,\kappa)=\frac{\omega}{\sqrt{2\pi}\kappa}e^{-\frac{\omega^2}{8\kappa^2}(4x'^2+y'^2)}\left(\cos(\omega x')-e^{-\frac{\kappa^2}{2}}\right), \qquad (1)$$

where $x'=(x-x_0)\cos\theta+(y-y_0)\sin\theta$, $y'=-(x-x_0)\sin\theta+(y-y_0)\cos\theta$, (x_0, y_0) is the center of the function, ω is the radial frequency in radians per unit length, and θ is the orientation of the Gabor filters in radians. κ is defined as $\kappa=\sqrt{2\ln 2}\left(\frac{2^\delta+1}{2^\delta-1}\right)$, where δ is the half-amplitude bandwidth of the frequency response. The orientation of a local region is estimated using a competitive rule,

$$j=\arg\min_p \int\int I(x,y)\psi_R(x,y,x_o,y_o,\omega,\theta_p,\kappa)dxdy, \qquad (2)$$

where j is called the winning index. The orientations of the six filters, θ_p are $p\pi/6$, where p=0, 1, 2, 3, 4 and 5. Fig. 2(d) shows a Competitive Code.

4) The matcher: This component compares two Competitive Codes based on their angular distance defined as

$$A_f(P,Q)=\sum_{x=1}^{32}\sum_{y=1}^{32}A(P_{x,y},Q_{x,y}), \qquad (3)$$

where $P_{x,y}$ $(Q_{x,y})$ is a winning index of Competitive Code, $P(Q)$ at position (x, y) and $A(P_{x,y},Q_{x,y})$ is the angular distance between the two winning indexes listed in Table 1. We design a coding scheme, given in Table 2, to encode the winning indexes for real-time identification. The corresponding bitwise angular distance is defined as

$$A_f(P,Q)=\sum_{x=1}^{32}\sum_{y=1}^{32}\sum_{i=1}^{3}P_i^b(x,y)\otimes Q_i^b(x,y), \tag{4}$$

where $P_i^b(Q_i^b)$ is the i^{th} bit plane of Competitive Code *P(Q)* and $\otimes$ is bitwise exclusive OR. Occasionally, some palmprints contain non-palmprint pixels because of the incorrect placement of hands. A simple threshold can be used to classify the non-palmprint pixels since they belong to a capture device having low gray levels. A bit plane is used as a mask to denote the non-palmprint pixels. Finally, the bitwise angular distance is defined as

$$A_f(P,Q)=\frac{\sum_{y=1}^{32}\sum_{x=1}^{32}\sum_{i=1}^{3}(P_M(x,y)\cap Q_M(x,y))\cap(P_i^b(x,y)\otimes Q_i^b(x,y))}{3\sum_{y=1}^{32}\sum_{x=1}^{32}P_M(x,y)\cap Q_M(x,y)}, \tag{5}$$

where $\cap$ is bitwise AND and $P_M(Q_M)$ is the mask of *P(Q)*. The range of A_f is between 0 and 1 and the angular distance is zero for perfect matching. Twenty-five translated Competitive Codes are computed for alignment imperfections. Thus, 25 angular distances are obtained when two palmprints are matched. The minimum of these distances is considered as the final angular distance, A_F.

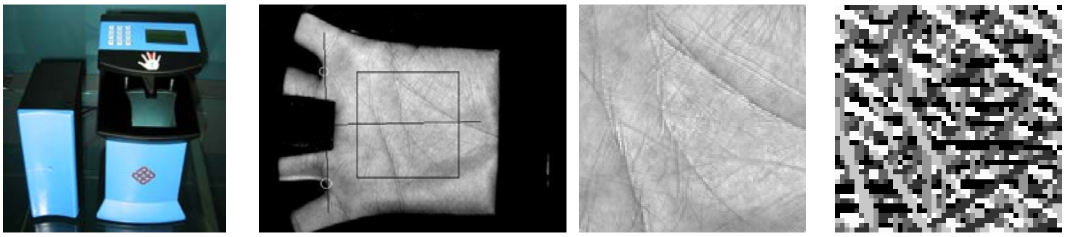

Fig. 2 Illustration of a palmprint identification system using Competitive Code. a) A palmprint scanner, b) the key points and coordinate system for palmprint segmentation and alignment, c) a preprocessed image and d) a Competitive Code, where different colors represent different orientations.

Table 1 All possible angular distances between different winning indexes

Angular Distance $A(P_{x,y}, Q_{x,y})$		Winning indexes, $P_{x,y}$					
		0	1	2	3	4	5
Winning indexes $Q_{x,y}$	0	0	1	2	3	2	1
	1	1	0	1	2	3	2
	2	2	1	0	1	2	3
	3	3	2	1	0	1	2
	4	2	3	2	1	0	1
	5	1	2	3	2	1	0

Table 2 Bitwise representation of the Competitive Code

Winning indexes	Bit 1	Bit 2	Bit 3
0	0	0	0
1	0	0	1
2	0	1	1
3	1	1	1
4	1	1	0
5	1	0	0

3. SECURITY MEASURES

3.1. Template Re-issuance

To generate cancellable Competitive Code, we insert a random orientation field into the feature extractor to generate noise-like feature codes. Thus, the feature extractor in Eq. 2 becomes

$$j = \arg\min_{p} \int\int I(x,y)\psi_R(x,y,\omega,\theta_p{+}\alpha(x_o,y_o),\kappa)dxdy\,, \tag{6}$$

where $\alpha(x_0, y_0) \in \{0, \pi/6, 2\pi/6, 3\pi/6, 4\pi/6, 5\pi/6\}$ is a random field. Thus, the six filters form a random orientation filter bank (ROFB). This random field would not degrade the original performance of Competitive Code since it does not affect the angular distance between two winning indexes. However, this random field does not generate non-invertible cancellable biometrics. It is noted that non-invertible cancellable biometrics tends to reduce accuracy [29]. Fig. 3 shows an original and two corresponding cancellable Competitive Codes. We should mention that similar ideas are employed for cancellable iris and for generating cryptographic key from biometrics [26-27].

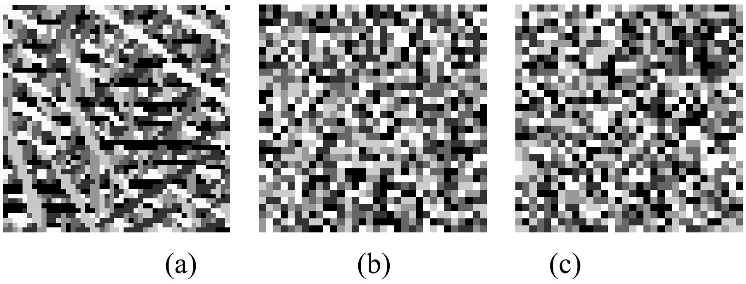

Fig. 3 Illustration of cancellable Competitive Codes. (a) Original Competitive Code and (b)-(c) Competitive Codes from different random fields.

3.2. Replay attacks

Although cryptography is not suitable for protecting the whole system, it is good for defending against replay attacks at Point 4 since only several templates are required to encrypt and to decrypt in each identification process. To protect this data link, we use a one time pad, which in cryptography is regarded as perfect secrecy. We generate random bits, R with the same size as Competitive Code, P^B (bitwise representation). Mathematically, the encryption process is $P^B \otimes R$ and the decryption process is $P^b \otimes R \otimes R$. It should be remembered that the random bits R are used one time only.

3.3. Database attacks

When carrying out database attacks, attackers will either insert unregistered templates or modify the templates in the database. So that the system can detect these attacks, the templates contain embedded secret messages. If each user provides only one template in the database, a part of winning indexes is changed to specific values and this acts as the secret message. A secret code is defined as (x_i, y_i, v_i), where i=1,…,m and m is the length of the secret message. The original winning indexes at the position, (x_i, y_i) are replaced by v_i i.e. $P(x_i, y_i) = v_i$. It is important to note that different messages are embedded in different templates. Otherwise, attackers can uncover the messages easily.

If each user has more than one template in the database, the correlation between templates from the same user can be used to extract the messages. Let P_1 and P_2 be two templates from the same palm; S_1 and S_2 be two messages and P_{s1} and P_{s2} be two templates with messages. The error map of P_{s1} and P_{s2} is defined as $e(x, y) = A(P_{s1}(x, y), P_{s2}(x, y))$, where A is defined in Table 1. The error map shows the angular distances between winning indexes at each location. If P_1 and P_2 are nearly identical, i.e. $P_1 \approx P_2$ and we hide two totally different messages, i.e. $S_1 \neq S_2$ in them to generate P_{s1} and P_{s2}, the messages have a high probability in the positions where $e(x, y) \neq 0$. If P_1 and P_2 are very different and we hide an identical message, i.e. S_1=S_2 in them to generate P_{s1} and P_{s2}, the messages have to be in the positions where $e(x, y) = 0$. Therefore, we need a more complex scheme to hide the secret messages. We design secret messages with two parts, S_{ID}=(x_{IDi}, y_{IDi}, v_{IDi}) and S_T =(x_{Ti}, y_{Ti}, v_{Ti}), where the subscript ID represents the message depending on a user identity and the subscript T

represents the message depending on a template identity. All the templates from the same palm are embedded the same S_{ID}. However, their S_Ts are different. To hide S_{ID} in k templates P_0, P_1, P_2…and P_k from the same palm, one of template P_0 is selected for aligning other templates and the vertical and horizontal translations (h_j, v_j) between P_0 and P_j are computed. To take into account the translation, we set $P_j(x_{IDi}+h_j, y_{IDi}+v_j)=v_{IDi}$ for S_{ID} while we set $P_j(x_{Ti}, y_{Tj})=v_{Ti}$ for S_T.

To further hide the correlation between the templates from the same palm, random bits R_c are used to encrypt the c^{th} templates of all users, i.e. $P^B \otimes R_c$. Although R_c is not a one-time pad, not perfect secrecy, it raises the level of difficulty involved in using the correlation to carry out database attacks. Some may immediately think that we need to decrypt the templates for matching. It should be noted that decrypting input Competitive Code, Q^B by R_c i.e. $Q^B \otimes R_c$ and decrypting templates in the database are equivalent activities and the number of R_c which is equal to number of templates per user is limited. Therefore, the computation cost would not be drastically increased. We also hide secret messages in input Competitive Codes to detect replay attacks at Point 4.

4. EXPERIMENTAL RESULTS

Although longer messages provide greater security, they can also degrade the recognition performance. In this section, we examine the accuracy of Competitive Code with message lengths of, 0, 8, 16, 24 and 32 winning indexes.

We collected palmprint images from 284 subjects. In this dataset, 98 subjects are female. Of the total number of subjects, 89%, are younger than 30 years old, 1% are older than 50 and about 10% are aged between 30 and 50. The palmprint images were collected on two separate occasions for reliable performance evaluation [33]. The average time interval between the first collection and second collection was 73 days. The minimum and maximum time intervals were one day and 340 days, respectively. On each occasion, each subject was asked to provide about 10 images, each of the left palm and the right palm. Therefore, our database contains 11,074 palmprint images from 568 different palms. The image size is 384×284 and their resolution is 75 dpi.

To reliably estimate the accuracy of Competitive Code with different messages, each of the palmprint images was compared only with all of the palmprint images in the database from different occasions. A matching is considered as a genuine matching if two palmprints are from the same palm. Otherwise, it is considered as an imposter matching. The number of genuine matchings and imposter matchings in each experiment are 54,081 and 30,603,388, respectively. The genuine and imposter distributions are estimated by the genuine and imposter matchings and their corresponding receiver operating characteristic (ROC) curves are given in Fig. 4. Fig. 4 illustrates that the degradation of performance is not serious even when the message length is 32 winning indexes.

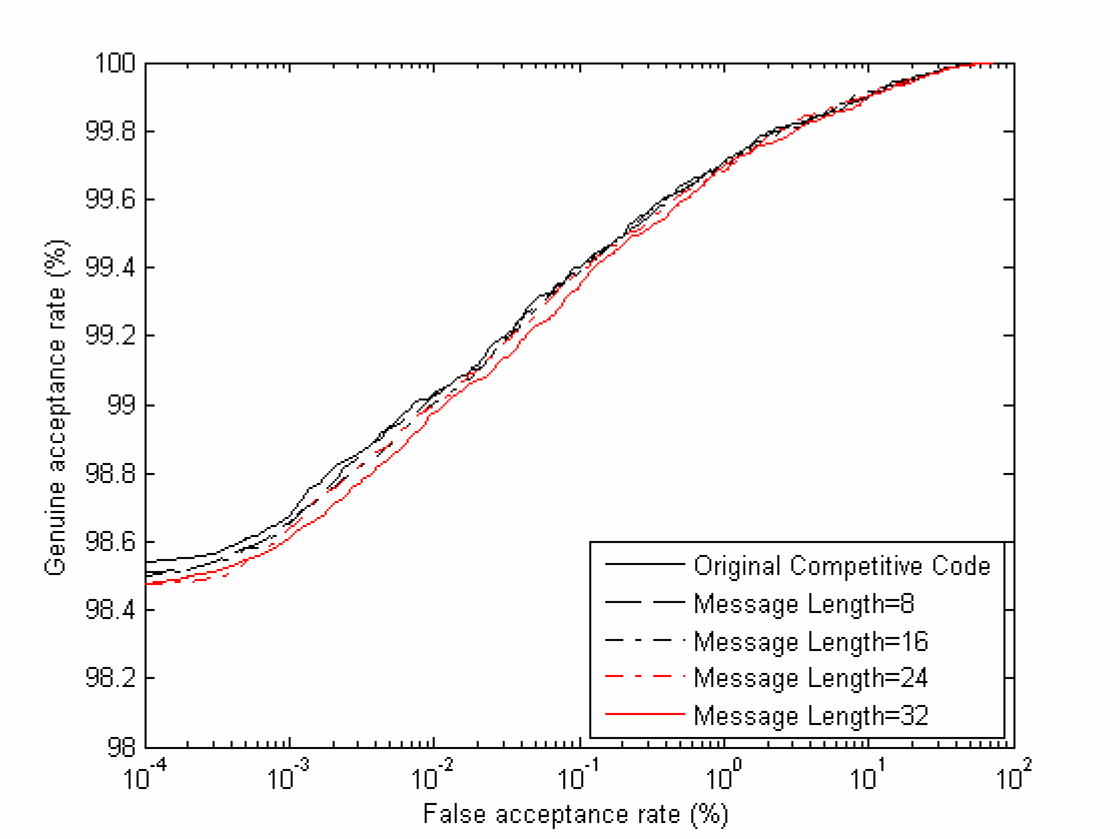

Fig. 4 Comparison of Competitive Codes with different lengths of messages.

5. ANALYSIS OF THE SECURITY MEASURES

In this section, we analyze the security measures to evaluate their effectiveness. We first estimate the number of effective cancellable templates. We then develop probability models to compute the probability of break-ins using database attacks. We do not consider replay attacks at Point 4 since a one time pad is regarded as perfect secrecy.

5.1. Ability of template re-issuance

The ROFB can generate $6^{1024} \approx 10^{797}$ different Competitive Codes for one single image. Although this number is extremely large, not all the templates can be used for re-issuances. The angular distance between compromised and reissued templates can be shorter than the decision threshold. Since the random fields are independent, matching compromised templates and reissued templates is equivalent to matching two random synthetic templates. Therefore, we can apply the projected multinomial distribution developed for analyzing brute-fore attacks to estimate the probability of the angular distance between compromised and reissued templates being shorter than a threshold *T* [28]. The projected multinomial distribution is defined as,

$$\Pr(A_f(P,Q)=T)=\sum_{W \ni WK^T=T} f(w_o,w_1,w_2,w_3)\,, \tag{7}$$

where

$$f(w_0,w_1,w_2,w_3)=\frac{n!}{w_0!w_1!w_2!w_3!}p_0^{w_0}p_1^{w_1}p_2^{w_2}p_3^{w_3}\,, \tag{8}$$

w_i is the number of $A(P_{x,y},Q_{x,y})=i$, $W=[w_o,w_1,w_2,w_3]$, $K=[0,1,2,3]$, p_i is the probability of $A(P_{x,y},Q_{x,y})=i$ and n=1024. The derivation of the distribution can be found at [28]. According to this theoretical probabilistic model, when the threshold is 0.39, which is generally used in the system, we can reissue 10^{15} templates and corresponding probability of using compromised templates to break in is 10^{-12} [28]. These numbers demonstrate that the probability of a successfully break-in into the system using compromised templates is extremely low and the number of effective cancellable templates is numerous.

5.2. Analysis of database and replay attacks

To perform database attacks, attackers firstly need to estimate the random field and the random bits R_c

to adjust their Competitive Codes from unregistered users, which can be generated based on our publications. Then, they either insert the adjusted templates directly into the database or combine the unregistered template with registered templates to form a new template and insert it into the database. So far, the authors do not have any effective approaches for estimating the random field and R_c. If the entropies of winning indexes are low, it is easy for attackers to estimate them. The entropy at point (x, y) is defined as $E(x,y)=\sum_{i=0}^{5}-\Pr(P_{x,y}=i)\log(\Pr(P_{x,y}=i))$. The log is base 2 in our experiments. Figs. 5 (a) and (b) show the entropies of winning indexes of left and right palms, respectively. One thousand and two hundred images from 150 different palms are used to estimate the entropies in each figure. The lowest entropies of both left and right palms are 2.4. Comparing the maximum entropy bound, $-\log(1/6)=2.6$, they are still high.

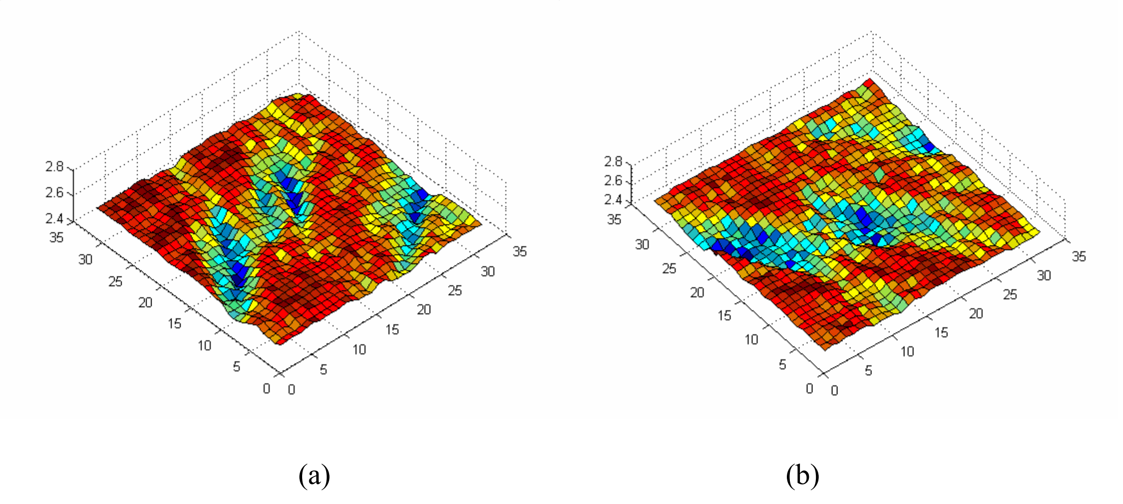

Fig. 5 The entropies of winning indexes. (a) left palm and (b) right palm

In the following analyses, we study the probabilities of an insert template and a registered template having the same secret messages assuming that attackers can perfectly estimate the random field and the random bits R_c. If attackers replace a registered template with an unregistered template directly, the probability of the two templates has the same message is 6^{-m}, where m is length of the message. For m=8, 16, 24 and 32, the corresponding probabilities are 5.9×10^{-7}, 3.5×10^{-13}, 2.1×10^{-19} and 1.3×10^{-25}, respectively.

Attackers may use a more intelligent approach to increase the probability of successful break-ins. They can select some winning indexes from a registered template and the others from an unregistered template to form a new template for database attacks. Let us consider a simple case first. Each user has only one template in the database. Let the number of winning indexes selected from the registered template be M and the number of message codes being selected be u. In other words, the number of winning indexes from the unregistered template is 1024-M. If we consider u as a random variable, the probability of the M winning indexes containing r secret codes can be computed by hypergeometric distribution, whose probability density function is defined as,

$$h(r) = \frac{{}_{m}C_{r}\ {}_{1024-m}C_{M-r}}{{}_{1024}C_{M}}. \tag{9}$$

Therefore, the probability of the combined template and the registered template have the same secret message is $\sum_{r=0}^{m} h(r)/6^{m-r}$. Table 3 lists the numerical values.

Table 3 The probability of combined and registered templates having the same secret message in the case that a system stores only one template.

Number of winning indexes in combined template is drawn from registered templates	Message length, *m*			
	8	16	24	32
256	3.8×10^{-4}	1.4×10^{-7}	4.7×10^{-11}	1.5×10^{-14}
512	1.3×10^{-2}	1.7×10^{-4}	2.1×10^{-6}	2.5×10^{-8}

If each user has more than one template, attackers can make use of the correlation between the templates of the same user to further increase the probability of successful break-ins. First of all, they can align all templates based on one of the templates and compute the error map defined as $e(x,y)=\sum_{i=1} A(P_{s0}(x,y),P_{si}(x+h_i,y+v_i))$. The locations of the message S_{ID} are in the zeros of error map and between $2<x\le 30$ and $2<y\le 30$ because of the translations. Let the number of zeros between $2<x\le 30$ and $2<y\le 30$ be *z*. If attackers select all the corresponding winning indexes from P_{s0}, the combined templates have to include $S_{ID.}$ Some secret codes of S_T are also included in this selection process. Let the number of secret codes of S_T being selected is *t*. The sizes of S_{ID} and S_T both are set to *m*/2 in this analysis. Thus, the number of secret codes that were not selected is *m/2-t*. Since attackers have selected *z* winning indexes, the number of selection for the rest of unselected secret codes is *M-z* and the number of rest of winning indexes in P_{s0} is 1024-*z*. The probability of *r* secret codes being selected from the rest of winning indexes is

$$g(r)=\frac{{}_{m/2-t}C_{r}\ {}_{1024-z-m/2+t}C_{M-z-r}}{{}_{1024-z}C_{M-z}}. \tag{10}$$

Thus, the probability of the combined template and registered template having the same message code is $\sum_{r=0}^{m/2-t} g(r)/6^{m/2-t-r}$. In addition to the parameters *M* and *m*, we need the other two parameters, *t* and *z* to compute this probability.

In the following experiments, the system stores three templates for each palm to estimate *t* and *z*. All the templates are from palmprints collected in the first session. We repeat the experiment 10 times by randomly selecting palmprints from the first session to construct the registered templates. In total, we have 5,680 sets of *z* and *t*. Tables 4(a) and (b) report the average and maximum probabilities of combined templates and registered templates having the same message, respectively. Tables 3 and 4 demonstrate that attackers can make use of the correlation between templates to increase the probability of successful break-ins. However, the probabilities are still very low if the message length is longer than or equal to 32.

We should remember that database attacks on biometric systems are in fact different from cipher attacks. For cipher attacks, attackers have encrypted data and try to recover the original data, such as text. Generally speaking, they can perform unlimited attempts and know the correctness of decrypted data. However, attackers of biometric systems have only very limited number attempts available to them in which to carry out database attacks since they need to insert the combined templates into databases in order to examine them for correctness.

Table 4 The average (a) and maximum (b) probability of combined and registered templates having the same secret message in the case that system stores only three templates.

(a)

Number of winning indexes in combined template is drawn from registered templates	Message length, *m*			
	8	16	24	32
256	2.8×10^{-4}	1.2×10^{-6}	5.8×10^{-9}	4.2×10^{-11}
512	2.9×10^{-2}	1.0×10^{-3}	4.6×10^{-5}	1.9×10^{-6}

(b)

Number of winning indexes in combined template is drawn from registered templates	Message length, *m*			
	8	16	24	32
256	5.0×10^{-2}	4.8×10^{-4}	3.3×10^{-6}	1.4×10^{-7}
512	3.7×10^{-1}	5.7×10^{-2}	3.1×10^{-3}	1.9×10^{-4}

6. OPEN PROBLEMS IN BIOMETRIC SECURITY

Currently, most biometric systems are examined only on zero effort attacks (general false acceptance rates). Do they still survive if experts attack them? In addition, how can we objectively evaluate and compare the security of biometric systems? We should bear in mind that successfully breaking into biometric systems requires money, time and knowledge. Biometric researchers are facing a dilemma. Of course we have a scientific responsibility to publish papers so as to disclose our findings and algorithms to distribute knowledge to society and to build the future base of our fields. We must be aware, however, that at the same time we are also providing potential attackers with the information that they require to break into our systems.

7. CONCLUSION

In this paper, we employ random orientation field banks for template re-issuance, one time pads for defending replay attacks and secret messages for detecting replay and database attacks. The analysis demonstrates that random orientation field banks can re-issue numerous templates and secret messages can detect database and replay attacks effectively if the message length is longer than or equal to 32. The experimental results also show that the presence of messages in the templates results in a low degradation of accuracy.

REFERENCES:

[1] A. Jain, R. Bolle and S. Pankanti (eds.), Biometrics: Personal Identification in Networked Society, Boston, Mass: Kluwer Academic Publishers, 1999.

[2] J.O. Kim, W. Lee, J. Hwang, K.S. Baik and C.H. Chung, “Lip print recognition for security systems by multi-resolution architecture”, Future Generation Computer Systems, 20 (2) (2004) 295-301.

[3] A.K. Jain, S.C. Dass and K. Nandakumar, “Soft biometric traits for personal recognition systems”, in Proceedings of International Conference on Biometric Authentication, (2004) 731-738.

[4] S.A. Israel, J.M. Irvine, A. Cheng, M.D. Wiederhold and B.K. Wiederhold, “ECG to identify individuals”, Pattern Recognition, 38 (1) (2004) 133-142.

[5] C.C Han, “A hand-based personal authentication using a coarse-to-fine strategy”, Image and Vision Computing, 22 (11) (2004) 909-918.

[6] D. Zhang, W.K. Kong, J. You and M. Wong, “Online palmprint identification”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 25 (9) (2003) 1041-1050.

[7] C.C. Han, H.L. Cheng, C.L. Lin and K.C. Fan, “Personal authentication using palm-print features”, Pattern Recognition, 36 (2) (2003) 371-381.

[8] G. Lu, D. Zhang and K. Wang, “Palmprint recognition using eigenpalms features”, Pattern Recognition Letters, 24 (9) (2003) 1463-1467.

[9] A. Adler, “Sample images can be independently restored from face recognition templates” in Proceedings of Canadian Conference on Electrical and Computer Engineering, Montreal, Canada, 2003 1163-1166.

[10] X. Wu, D. Zhang, K. Wang and B. Huang, “Palmprint classification using principal lines”, *Pattern Recognition*, 37 (10) (2004) 1987-1998.

[11] L. Zhang, D. Zhang, “Characterization of palmprints by wavelet signatures via directional context modeling”, IEEE Transaction on System Man and Cybernetics, Part B, 34 (3) (2004) 1335-1347.

[12] J. You, W.K. Kong, D. Zhang, K.H. Cheung, “On hierarchical palmprint coding with multiple features for personal identification in large databases”, IEEE Transactions on Circuit Systems for Video Technology, 14 (2) (2004) 234-243.

[13] S. Ribaric, D. Ribaric and N. Pavesic, “Multimodal biometric user-identification system for network-based applications”, IEE Proceedings, Vision, Image and Signal Processing, 150 (6) (2003) 409-416.

[14] W. Li, D. Zhang, Z. Xu, “Palmprint identification by Fourier transform”, International Journal of Pattern Recognition and Artificial Intelligence, 16 (4) (2003) 417-432.

[15] Y.H. Pang, A. Teoh, D. Ngo, H.F. San, “Palmprint verification with moments”, Journal of Computer Graphics, Visualization and Computer Vision, 12 (2) (2004) 325-332.

[16] X.Y. Jing and D. Zhang, "A face and palmprint recognition approach based on discriminant DCT feature extraction", IEEE Transaction on System Man and Cybernetics, Part B, 34 (6) (2004) 2405-2415.

[17] A. Kong, D. Zhang and M. Kamel, "Palmprint identification using feature-level fusion", Pattern Recognition, 39 (3) (2006) 478-487.

[18] A.W.K. Kong and D. Zhang, "Competitive Coding scheme for palmprint verification", in Proceedings of International Conference on Pattern Recognition, 1 (2004) 520-523.

[19] N.K. Ratha, J.H. Connell and R.M. Bolle, "Enhancing security and privacy in biometrics-based authentication systems", IBM Systems Journal, 40 (3) (2001) 614-634.

[20] N.K. Ratha, J.H. Connell and R.M. Bolle, "Biometrics break-ins and band-aids", Pattern Recognition Letters, 24 (13) (2003) 2105-2113.

[21] T. Connie, A. Teoh, M. Goh, "PalmHashing: a novel approach for dual-factor authentication", Pattern Analysis and Applications, 7 (3) (2005) 255-256.

[22] T. Connie, A. Teoh, M. Goh and D. Ngo, "PalmHashing: a novel approach for cancellable biometrics", Information Processing Letters, 93 (1) (2005) 1-5.

[23] A. Kong, K.H. Cheung, D. Zhang, M. Kamel and J. You, "An analysis of Biohashing and its variants", Pattern Recognition, 39 (7) (2006) 1359-1368.

[24] U. Uludag and A.K. Jain, "Attacks on biometric systems: a case study in fingerprints", in Proceedings of SPIE-EI 2004, Security, Seganography and Watermarking of Multimedia Contents VI, 2004 622-633.

[25] U. Uludag, S. Pankanti, S. Prabhakar and A.K. Jain, "Biometric cryptosystems: issues and challenges", Proceedings of the IEEE, 92 (6) (2004) 948-960.

[26] M. Braithwaite, U.C. von Seelen, J. Cambier, J. Daugman, R. Class, R. Moore and I. Scott, "Applications-Specific Biometric Template", IEEE Workshop on Automatic Identification Advanced Technologies, Tarrytown, NY, March, 14-15, 2002 167-171.

[27] C. Soutor, D. Roberge, S.A. Stojanov, R. Gilroy and B.V.K. Vijaya Kumar, "Biometric encryption", in ICAS Guide to Cryptography, R.K. Nichols, Ed. New York: McGraw-Hill, 1999.

[28] A. Kong, D. Zhang and M. Kamel, "An analysis of brute-force break-ins of a palmprint authentication system", IEEE Transactions on Systems, Man and Cybernetics, Part B, 36 (5) (2006) 1201-1205.

[29] K.H. Cheung, A. Kong, D. Zhang, M. Kamel, J. You and T.H.W. Lam, "An analysis on accuracy of Cancellable biometrics on Biohashing", in Proceeding of the 9th International Conference on Knowledge-based intelligent information and engineering system, 2005 1168-1172.

[30] M. Savvides, B.V.K. Vijaya Kumar and P.K. Khosla, "Cancellable biometric filters for face recognition", in Proceedings of the 17th International Conference on Pattern Recognition, 3 (2004) 922-925.

[31] A.T.B. Jin, A. Goh and D.C.L. Ngo, "Random Multispace Quantization as an Analytic

Mechanism for BioHashing of Biometric and Random Identity Inputs", IEEE Transactions on Pattern Analysis and Machine Intelligence, 28 (12) (2006) 1892-1901.

[32] D. Hu, G. Feng, and Z. Zhou, "Two-dimensional locality preserving projections (2DLPP) with its application to palmprint recognition", Pattern Recognition, 40 (1) (2007) 339-342.

[33] H.K. Cheung, A. Kong, D. Zhang, M. Kamel and J. You, "Does EigenPalm work? A system and evaluation perspective", in Proceedings of International Conference on Pattern Recognition 4 (2006) 445-448.