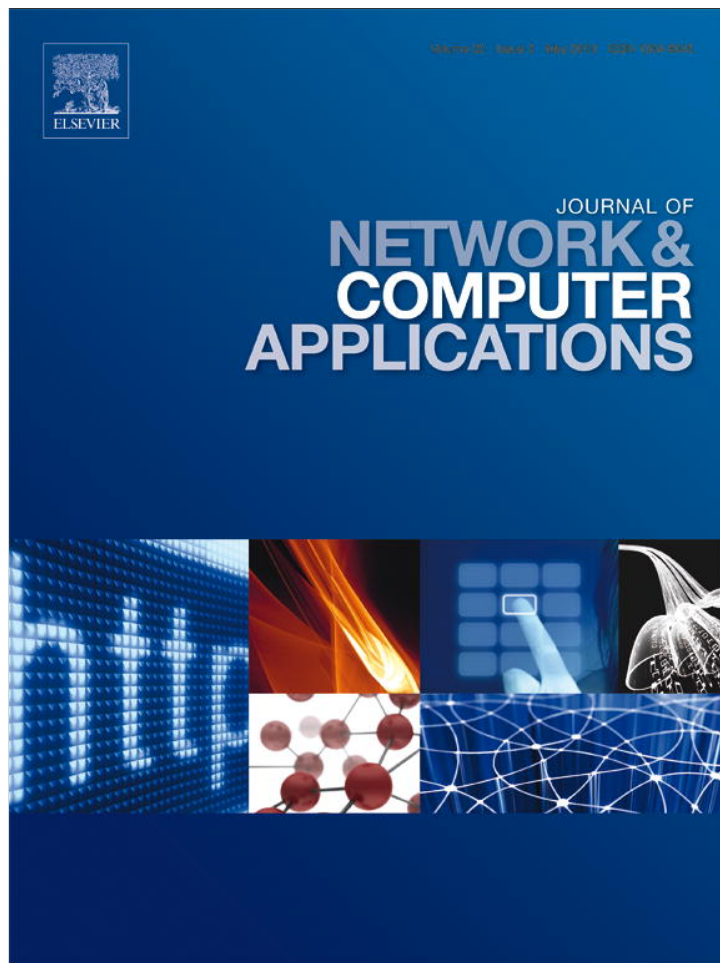


Provided for non-commercial research and education use.  
Not for reproduction, distribution or commercial use.



This article appeared in a journal published by Elsevier. The attached copy is furnished to the author for internal non-commercial research and education use, including for instruction at the authors institution and sharing with colleagues.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier's archiving and manuscript policies are encouraged to visit:

<http://www.elsevier.com/copyright>



Contents lists available at ScienceDirect

## Journal of Network and Computer Applications

journal homepage: [www.elsevier.com/locate/jnca](http://www.elsevier.com/locate/jnca)

## Minutiae and modified Biocode fusion for fingerprint-based key generation

Eryun Liu<sup>a,b</sup>, Jimin Liang<sup>a,\*</sup>, Liaojun Pang<sup>c</sup>, Min Xie<sup>c</sup>, Jie Tian<sup>a,d</sup><sup>a</sup> Life Sciences Research Center, School of Life Sciences and Technology, Xidian University, Xian, Shaanxi 710071, China<sup>b</sup> School of Electronic Engineering, Xidian University, Xian, Shaanxi 710071, China<sup>c</sup> Ministry of Education Key Laboratory of Computer Network and Information Security, School of Telecommunication Engineering, Xidian University, Xian, Shaanxi 710071, China<sup>d</sup> Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

## ARTICLE INFO

## Article history:

Received 28 July 2009

Received in revised form

15 November 2009

Accepted 7 December 2009

## Keywords:

Key generation

Secure sketch

Feature fusion

Fuzzy extractor

Fingerprint alignment

## ABSTRACT

Key generation from biometrics has been studied intensively in recent years, linking a key with certain biometric enhances the strength of identity authentication. But the state-of-the-art key generation systems are far away from practicality due to low accuracy. The special manner of biometric matching makes a single feature based key generation system difficult to obtain a high recognition accuracy. Integrating more features into key generation system may be a potential solution to improve the system performance. In this paper, we propose a fingerprint based key generation system under the framework of fuzzy extractor by fusing two kinds of features: minutia-based features and image-based features. Three types of sketch, including minutiae based sketch, modified Biocode based sketch, and combined feature based sketch, are constructed to deal with the feature differences. Our system is tested on FVC2002 DB1 and DB2, and the experimental results show that the fusion scheme effectively improves the system performance compared with the systems based only on minutiae or modified Biocode.

© 2009 Elsevier Ltd. All rights reserved.

## 1. Introduction

Modern cryptographic system security is solely based on a sufficiently long key while the details of encryption/decryption algorithms are public to everyone (Stallings, 2005). In the modern cryptographic system, a secure channel is supposed to exist for distributing encryption key, such as storing the key in a smart card which belongs to the legitimate user or being managed by a centralized server (e.g., Certificate Authority, CA) where a password is used to control the key access. It is well known that smart card suffers from being lost or stolen, and the password can be easily forgotten or guessed. The channel supposed to be secure is actually unsecured in practice. The difficulty is that, in a typical encryption system, the key has no strong relation with the user. The system just grants access to the people who has the right key, not the people whom the key belongs to. In another word, the system has no ability to distinguish a legitimate user from an imposter.

In recent years, the techniques of generating a reliable cryptographic key from biometrics have been studied (Juels and Wattenberg, 1999; Dodis et al., 2004; Juels and Sudan, 2006). In such a key generation system, the key is linked with biometrics which represent the physical identity of people, and not stored

explicitly. When it is needed, the key can be recovered from the same biometrics in real time and destroyed after being used. The key's existing periods is shortened as small as possible. In the new model of key management, the assumption that the channel for distributing key is secure has been removed. Some public data which leak few information about the key and biometric template are stored to help recovering the key.

Dodis et al. (2004) provide a framework of how to generate cryptographic key from biometric data. A refined work can be found in Dodis et al. (2008). In their work, two primitives, secure sketch and fuzzy extractor, are defined. The secure sketch describes how to recover the template biometric data from a closed resampled version of the same biometric by publishing the sketch data which leak few information about the template. Fuzzy commitment (Juels and Wattenberg, 1999) scheme and fuzzy vault scheme (Juels and Sudan, 2002) are two special cases of the secure sketch. The fuzzy extractor guarantees reliable generation of a cryptographic key from biometric data based on secure sketch construction. Whether the key has been recovered correctly or not indicates biometric matching success or failure. Fuzzy extractor techniques facilitate the key management in a cryptosystem.

However, another problem arises in the key generation system. It is difficult to construct a secure sketch for biometric features whose similarity measures are complex. For example, in a fingerprint recognition system, the synthetic similarity score is usually calculated by an equation which considers both local features (e.g., minutiae) and global features (e.g., orientation field). It is hard to measure such similarity in a secure sketch,

\* Corresponding author. Tel.: +86 29 88202649.

E-mail addresses: jimleung@mail.xidian.edu.cn, jimminliang@gmail.com (J. Liang).

hence the features used in the secure sketch are limited, which lead to low genuine accept rate (GAR) in a key generation system. How to fuse multi-features in a secure sketch based key generation system is still an open problem. Though it is difficult to include multi-features in single secure sketch, it is possible to construct a secure sketch for each feature and combine all the sketches together to generate a key. With this understanding, feature fusion approaches for key generation are proposed.

Fingerprint as a widely used biometric modality, has many shares of market for it is convenient to capture and accurate in practice. Many types of features can be extracted from fingerprint, e.g., minutiae (Tico and Kuosmanen, 2003), fingercode (Jain et al., 2000), orientation field (Kulkarni et al., 2006), and wavelet fourier mellin transform (WFMT) feature (Jin et al., 2004b). Fingerprint features can be roughly classified into two categories: minutia-based and image-based. Minutia-based features are the most widely used features in the literatures (Maltoni et al., 2009) due to the great discriminating ability. Image-based features also gain lots of attentions (Nanni and Lumini, 2009) thanks to its better capability in dealing with low quality images and fixed length representation. In Jain et al. (2000), the authors proposed an image-based feature representation called FingerCode. The problem of this approach is that it has to extract two variants of FingerCode to complement to alignment errors. Jin et al. (2004b) proposed an image-based feature by cropping a region of interesting (ROI) with respect to the center point and performing the wavelet Fourier mellin transform. Nanni and Lumini (2009) made a good survey on image-based fingerprint matching techniques.

BioHashing as a biometric template protection algorithm based on image feature was first proposed by Jin et al. (2004a). Following Jin et al. (2004a), Kong et al. (2006) pointed out that the zero equal error rate (EER) performance concluded in Jin et al. (2004a) was under a hidden assumption that the token was secure anytime. Many works have been done to improve the performance of BioHashing in the case of token been stolen (Nanni and Lumini, 2006b, 2008b; Lumini and Nanni, 2007). In Nanni and Lumini (2008a), a new image-based feature by using local binary pattern was proposed and BioHashing was coupled with this feature for a hybrid fingerprint matcher. In their method, minutiae were used to align template and query images, where the aligning method is infeasible in a biometric cryptosystem, because the template minutiae are unavailable during verification.

Nandakumar et al. (2007) proposed a fully automatic fingerprint fuzzy vault system based on minutia feature only. In their method, the high curvature points as auxiliary information were used to perform image alignment which gave too much information to the attackers. Kotlarchyk et al. (2008) investigated the parameters used in fuzzy vault system by simulation study. They concluded that alignment was critical to fuzzy vault system. Although increasing the matching threshold can over this problem to some extent, as a result, more chaff points will be treated as true minutia points. Minutiae itself can be used to align two fingerprints before an image-based method (Ross et al., 2003; Nanni and Lumini, 2007), but such alignment method cannot be deployed in a biometric cryptosystem, because the template minutiae information is unavailable during verification. A concatenated error correction scheme was proposed by Hao et al. (2006) to combine a cryptographic key with a binary vector feature, iris code. Bringer et al. (2008) pointed out that the good performance obtained in Hao et al. (2006) due to the high quality of iris images and Bringer et al. (2008) also proposed a 2-D iterative min-sum decoding algorithm to improve the results of binary vector based secure sketches.

Although the fixed length feature vector can be applied to many secure sketch construction, such as above-mentioned concatenated error correction scheme, min-sum decoding, as pointed out by Nanni and Lumini (2009), none of the performance gained by the image-based matchers is comparable with that obtained by the best minutia-based matchers. Fusion of minutia-based features and image-based features provides a potential way to improve the overall performance. There are two important problems which have to be tackled in the feature fusion based fingerprint key generation system. The first problem is about feature selection. All the features used in the system should be complemented to each other. Another problem is how to fuse the features, measured in different ways, within the framework of secure sketch.

Nandakumar and Jain (2008) proposed a multi-biometric, fingerprint and iris, template security method using fuzzy vault. In their work, iriscodes were changed into unordered set by salting. Bose–Chaudhuri–Hocquenghem Code (BCH code), which was found by Bose and Ray-Chaudhuri (1960) and Hocquenghem (1959) and is a class of widely used error correction code (ECC), is used for the salting operation. The unordered set was combined with minutia set to construct a fuzzy vault. Though their multi-biometric system obtains a high genuine accept rate (GAR) when false accept rate (FAR) is on a low level, the performance of single biometric based system does not improved. Nagar et al. (2008) proposed a key binding system based on fuzzy vault and fuzzy commitment. Two kinds of features were extracted, minutia coordinates and their corresponding descriptors. Minutia descriptors were used in a fuzzy commitment scheme to secure ordinate values in the vault. Their system is actually a minutia-based implementation, no global features are used.

In this paper, we devise a new fingerprint-based key generation system. In our system, the minutia-based features and image-based features are fused under secure sketch, where the minutia features, image-based features, and the fused features are all protected by secure sketch. The matching procedures are performed between query features and their corresponding template sketch data. Three sketches are constructed for feature fusion and key generation. Firstly, a minutiae based sketch (MS) is constructed by adding sufficient randomly selected chaff points. Then, a Biocode based PinSketch (BS) is used for Biocode recovering. Finally, a combined sketch (CS) by using the improved Juels and Sudan fuzzy vault (IJS fuzzy vault) is used for key recovering. The key is generated from the fused feature by using the strong randomness extractor techniques. In order to perform accurate alignment, we also present a reliable core point direction extraction algorithm for aligning template fingerprint image and query image which is robust to noise and leaks few information about the template. Moreover, a modified Biocode algorithm is proposed. The random projection and thresholding procedure are redesigned for fingerprint Biocode extraction.

The rest of this paper is organized as follows: In Section 2, we describe the features used in our system, including a new core point direction extraction algorithm and a modified Biocode extraction algorithm. The proposed key generation is presented in Section 3. The experimental results are shown in Section 4. Security of the system are analyzed in Section 5 and conclusions are drawn in Section 6.

## 2. Feature extraction and transformation

### 2.1. Minutiae extraction

Given a fingerprint image  $I$ , the short time Fourier transformation (STFT) based enhancement algorithm proposed by Chikkerur

et al. (2007) is adopted to obtain an enhanced binary fingerprint image. STFT method decomposes the input fingerprint image into overlapped blocks. For each block, the orientation, frequency and mask information are obtained by short time Fourier analysis. Based on these information, Fourier domain contextual filtering is performed on each block. The enhanced fingerprint image is obtained by combining all the blocks after taken inverse Fourier transform. Locally adaptive thresholding method (Giuliao et al., 1977) is used to binarize the enhanced gray scale image. Then chain code based method (Shi and Govindaraju, 2006) is used to extract minutia points. First, trace all the ridges in the binary image, then, detect all ridge points which have a significant left turn or right turn, ridge points which fall into a local small region are treated as one minutia point by averaging the location and orientation. The minutia set is represented as  $M = \{m_i\}_{i=1}^N$ , where  $N$  is the number of minutiae in  $I$ .

## 2.2. Fingerprint alignment based on core point

The alignment of template and query fingerprint images is critical to filter out the chaff points in a fuzzy vault based system. Different to traditional fingerprint authentication systems, the alignment for fuzzy vault is performed in the encryption domain. In consideration of templates security, not too much of information is stored for alignment. The difficulty of this step is to achieve best aligning accuracy with very limited information.

Early implementations of fingerprint-based fuzzy vault assumed that the template and query fingerprint images are pre-aligned, which is unpractical (Clancy et al., 2003). Yang and Verbauwhe (2005) proposed a fuzzy vault implementation based on reference point. In Uludag and Jain (2006), the authors extract high curvature points from orientation field flow curve of template and query fingerprint images, and the parameters of translation and rotation between template and query images are estimated by aligning the two high curvature point lists with an ICP algorithm. This method was improved by Nandakumar et al. (2007). Chung et al. (2005) propose an automatic alignment technique by searching a pair of minutia between query minutiae and vault. In Li et al. (2008), the authors proposed to use some transformed minutiae around core point for alignment. However, the transformed minutiae still leak some information about their original version.

In this section, we present a new algorithm for reliable core direction extraction. The core's location and direction are used together to perform accurate alignment.

Singular points are global feature of fingerprint images and invariant to translation, rotation, enlargement, and shrinking (Fan et al., 2008). Because of these characteristics, singular points can be used for fingerprint indexing, as well as for fingerprint alignment and orientation field modeling. There are two types of singular points: core and delta. Fig. 1 illustrates the topological structure of core and delta points. In an actual fingerprint image, core points are usually more reliable and stable than delta points, because core points are mainly located at the center of a fingertip, while delta points are usually missed by large translation during scanning. Moreover, each core point has a dominant direction which can be used to determine the rotation between template and query images. Core points as global features leak few information about minutiae. Though there are algorithms which use the singular points as parameters to model the whole fingerprint orientation, original orientation field is needed to estimate the rest parameters of the model (Sherlock and Monro, 1993; Zhou and Gu, 2004). Compared with the existing alignment method for fuzzy vault, such as high curvature points used by Nandakumar et al. (2007), core point leaks less information about minutiae.

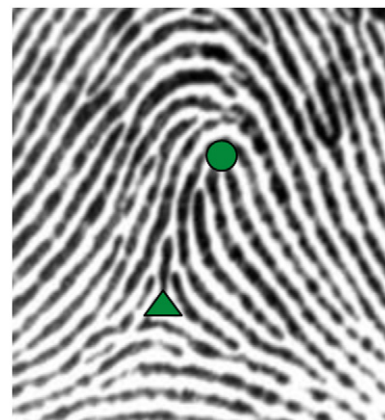


Fig. 1. The structure of singular points of fingerprint: core (●) and delta (▲).

Given a fingerprint image, our core point based alignment algorithm consists of the following steps: (A) orientation field extraction; (B) core point position and direction detection; and (C) minutiae transformation based on core point.

(A) *Orientation field extraction and core detection*: Accurate and smooth orientation field is essential to reliable core points detection and direction extraction. In recent years, many progresses on fingerprint orientation computation have been made (Wang et al., 2007; Ji and Yi, 2008; Huckemann et al., 2008; Ram et al., 2010). In this paper, we adopt FOMFE model described in Wang et al. (2007) to smooth orientation field and extract core points.

Firstly, the coarse orientation field is obtained by the gradient based method (Bazen and Gerez, 2002). Orientation in a block of size  $w \times w$  is perpendicular to the dominant gradient direction and can be calculated by

$$o(x, y) = \frac{1}{2} \tan^{-1} \frac{\sum_b 2G_x G_y}{\sum_b G_x^2 - G_y^2} + \frac{\pi}{2}, \quad (1)$$

where  $b$  is a block of size  $w \times w$  centered at  $(x, y)$ ,  $(G_x, G_y)$  is the gradient vector at  $(x, y)$ , and  $\tan^{-1}$  is a four-quadrant arctangent function.

Then, the orientation field is represented as a vector field  $(\cos 2o, \sin 2o)$ . Two functions,  $f_c(x, y)$  and  $f_s(x, y)$ , are used to approximate or fit  $\cos 2o$  and  $\sin 2o$ , respectively, by fourier series expansion, and the coefficients of series are obtained by linear least square optimization (LSQ). A continuum orientation field function can be obtained by

$$o'(x, y) = \frac{1}{2} \tan^{-1} \frac{f_c(x, y)}{f_s(x, y)} + \frac{\pi}{2}, \quad (2)$$

where  $x, y \in R$ .

The reconstructed fingerprint orientation field is very smooth. Singular points can be easily derived from the reconstructed orientation field. Considering the vector field of  $(\cos(2o'), \sin(2o'))$ , the gradient matrix at  $(x, y)$  can be obtained by

$$\mathbf{A}(x, y) = \begin{bmatrix} \frac{\partial \cos(2o')}{\partial x} & \frac{\partial \cos(2o')}{\partial y} \\ \frac{\partial \sin(2o')}{\partial x} & \frac{\partial \sin(2o')}{\partial y} \end{bmatrix}. \quad (3)$$

The singular points can be determined from  $\mathbf{A}$ . If  $\det|\mathbf{A}(x_c, y_c)| > Th_c$ , a core point is detected at  $(x_c, y_c)$ . On the contrary, if  $\det|\mathbf{A}(x_d, y_d)| < Th_d$ , a delta point is detected at  $(x_d, y_d)$ .  $Th_c$  and  $Th_d$  are predetermined threshold for core and delta point detection, respectively, and  $Th_c > 0$ ,  $Th_d < 0$ . If there are more than

one singular point in a small region, the average position will be taken instead.

In our fingerprint alignment method, only one core point is used, if there are more than one core point detected, the topmost one is used, and the rest singular points are ignored.

(B) *Core direction extraction*: The direction of a core point is actually ambiguous in an idea fingerprint orientation field. In our method, we define the direction of a core by the orientation flow curves through a neighboring region of this core point. The fingerprint orientation flow curves (FOFC) are more smooth and robust to noise than ridges (Dass and Jain, 2004). Fig. 2 shows three examples of core direction.

Given an orientation field  $o'(x,y)$  and a topmost core point  $c(x_c, y_c)$  which are obtained by FOMFE model, our core direction extraction procedure operates as follows (see Fig. 3).

The flow curves extraction algorithm is performed iteratively starting at the core point. To avoid ambiguousness of direction at the core point, we firstly start to trace the orientation flow curve at  $(x_c, y_c - \varepsilon)$ , where  $\varepsilon$  is a small positive integer (in our implementation,  $\varepsilon$  is set to be 5). For each side of the starting point  $(x_c, y_c - \varepsilon)$ , the tracing procedure operate iteratively until  $k$  points obtained or the boarder of image reached, the tracing step is set to be 4 pixels and  $k$  is set to be 100 in our implementation. The left-hand side flow curve and right-hand side flow curve are concatenated in a continuous sense. Then we get a continuous flow curve  $s = \{s_1, s_2, \dots, s_l\}$ , where  $l \leq 2k + 1$  is the length of  $s$  (see Fig. 4). The curvature map  $c$  of  $s$  is obtained by the method described in Nandakumar et al. (2007). Within a small neighboring region of the starting point, a point  $s_n$  with the maximum curvature value is found. The dominant direction vector of this flow curve is obtained by

$$\vec{v}_s = \frac{1}{2(N-1)} \left( \sum_{j=1}^N \frac{\vec{s}_n s_{n-j}}{|s_n s_{n-j}|} + \sum_{j=1}^N \frac{\vec{s}_n s_{n+j}}{|s_n s_{n+j}|} \right), \quad (4)$$

where  $N = \min\{l-n, n-1\}$ .

Suppose the coordinate of previously found high curvature point  $s_n$  is  $(x_n, y_n)$ , then the next starting point for tracing is set to

be  $(x_n, y_n - \varepsilon)$ . The searching procedure repeats until  $m$  flow curves extracted or the starting point reach the boarder of fingerprint image. Suppose  $m'$  flow curves' directions are extracted finally, where  $m' \leq m$ , the direction vector of the core point is obtained by

$$\vec{v} = \frac{1}{m'} \sum_{i=1}^{m'} \vec{v}_{s_i}, \quad (5)$$

where  $v_{s_i}$  is the  $i$  th flow curve's direction vector. The direction  $\theta_c$  of core point is the angle of  $\vec{v}$ .

*Alignment using core point*: If the position and direction of core points of template and query fingerprint images are both obtained, the alignment is a trivial task. Let  $(x_c^T, y_c^T, \theta_c^T)$  and  $(x_c^Q, y_c^Q, \theta_c^Q)$  be the core point of template and query images, respectively, and  $(x_m^Q, y_m^Q, \theta_m^Q)$  is a minutia of query fingerprint, then the alignment procedure is performed by

$$\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x_c^T \\ y_c^T \\ \theta_c^T \end{pmatrix} + \begin{pmatrix} \cos \Delta\theta & \sin \Delta\theta & 0 \\ -\sin \Delta\theta & \cos \Delta\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_m^Q - x_c^Q \\ y_m^Q - y_c^Q \\ \theta_m^Q - \theta_c^Q \end{pmatrix}, \quad (6)$$

where  $\Delta\theta = \theta_c^Q - \theta_c^T$  and  $(x', y', \theta')$  is the aligned coordinate of query minutia  $(x_m^Q, y_m^Q, \theta_m^Q)$ .

### 2.3. Modified Biocode extraction

Image-based fingerprint features are widely used in an automatic fingerprint identification system (AFIS) (Jain et al., 2000; Jin et al., 2004a; Kulkarni et al., 2006; Nanni and Lumini, 2006c). Image-based fingerprint identification method offers much higher computation efficiency with minimum pre-processing. In Jin et al. (2004b), Andrew proposed a method to extract features by applying wavelet Fourier-Mellin transform (WFMT) to a fingerprint image. The WFMT feature is invariant under translation and rotation. Based on this feature, Andrew further developed an algorithm called BioHashing (Jin et al., 2004a) to extract a binary code (also named Biocode) by performing inner product operation with a random orthogonal matrix. The original BioHashing

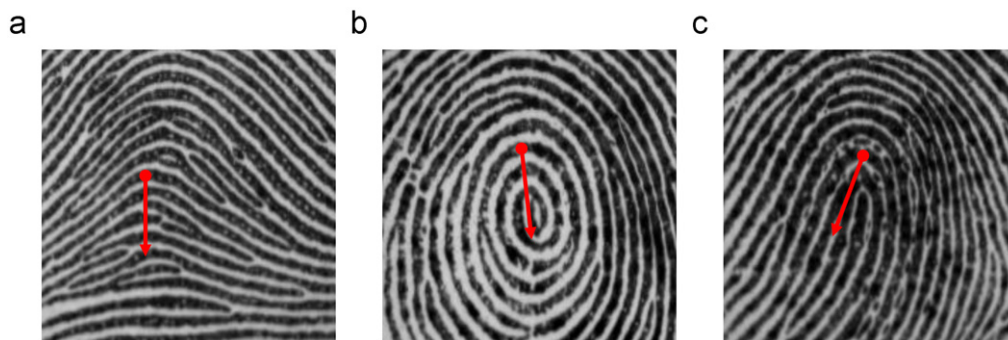


Fig. 2. Three examples of core direction marked manually.

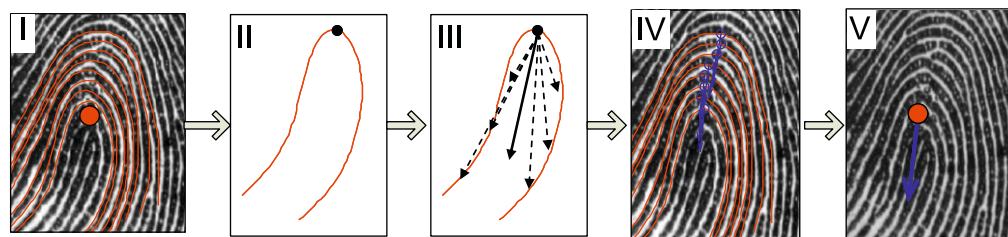


Fig. 3. Flow chart of core direction extraction procedure.

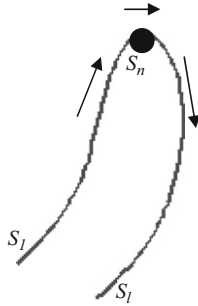


Fig. 4. One flow curve.

algorithm depends heavily on the token or random matrix. When token being stolen, the system performance of conventional Biocode dramatically degrades (Kong et al., 2006; Nanni and Lumini, 2006b).

In this section, we propose an improved BioHashing algorithm to extract Biocode for our key generation system. Lumini and Nanni (2007) found through experiments that the random projection and binarization are critical to BioHashing method. In our Biocode extraction procedure, the improvement is twofold: (1) the feature matrix of WFMT is projected into a lower dimension matrix by row projection and column projection, respectively. Such projection method can reserve most of the features during feature dimension reduction; (2) a thresholding vector is computed from database instead of zero vector which is used by Jin et al. (2004a). The thresholding vector is more suitable for the projected feature binarization.

Given a fingerprint image, the region of interesting of size  $n$  by  $n$  centered at core point is cropped. The core point detection algorithm is described in previous Section 2.2. Then an  $n$  by  $n$  WFMT feature matrix  $X$  is obtained by wavelet Fourier–Mellin transforming (Jin et al., 2004b). Based on WFMT feature  $X$ , Biocode extraction procedure consists of the following steps:

1. *Orthogonal random matrix generation*: Given a random seed  $s$ , generate a sequence of linearly independent row vectors  $r_i \in \mathfrak{R}^n$ , where  $i=1,2,\dots,m$ . The Blum–Blum–Shub (Blum et al., 1986) method is used as our pseudo-random bit generator. Apply the Gram–Schmidt ortho-normalization to transform the vectors  $r_i$  into an orthonormal set or vectors  $r'_i$ , where  $i=1,2,\dots,m$ . The row orthogonal vectors  $\{r'_i\}$  are used to form a row orthogonal matrix  $R=(r'_{ij})$ , where  $i,j$  indicate the row and column indices of  $R$ , respectively.
2. *Random projection*: A vector  $y$  is generated by concatenating the row vectors of  $Y$  which is obtained by
 
$$Y = RXR^T, \quad (7)$$
 where  $R^T$  is the transpose of  $R$ . The resultant  $Y$  is an  $m$  by  $m$  matrix, so,  $y$  is a vector of length  $m^2$ .
3. *Binarization*: A thresholding vector  $\{\tau_1, \tau_2, \dots, \tau_{m^2}\}$  is used to binarize  $y$  to obtain Biocode  $\{b_1, b_2, \dots, b_{m^2}\}$ :

$$b_i = \begin{cases} 0 & \text{if } y_i \leq \tau_i, \\ 1 & \text{if } y_i > \tau_i, \end{cases} \quad i = 1, 2, \dots, m^2, \quad (8)$$

where  $\{\tau_1, \tau_2, \dots, \tau_{m^2}\}$  is obtained by training  $L$  sample fingerprint images which are randomly selected from database. Use the previous steps, we can obtain  $L$  feature vectors  $\{y^k | k=1, 2, \dots, L\}$  of length  $m^2$  from the sample images. Let

$$\tau_i = \frac{1}{L} \sum_{k=1}^L y_i^k, \quad i = 1, 2, \dots, m^2. \quad (9)$$

The thresholding vector  $\{\tau_1, \tau_2, \dots, \tau_{m^2}\}$  is trained once and fixed for all users.

The random seed  $s$  will be stored in the smart card at encoding stage and the same seed will be used at decoding stage.

### 3. Fingerprint-based key generation system by fusing minutiae and modified Biocode

#### 3.1. Fundamental techniques

In this section, we will briefly describe some fundamental techniques which will be used later in this paper. In Dodis et al. (2004), the authors give a uniform framework for recovering biometric data, the secure sketch technique. Fig. 5 shows the flowchart of this framework, in which  $x$  is the template biometric data,  $p$  is public data of  $x$  named sketch produced by sketching procedure  $SS(\cdot)$  and  $p$  reveals few information about  $x$ , and  $x'$  is the query biometric data. When  $x$  and  $x'$  are close enough, the recovering procedure  $Rec(\cdot)$  guarantees the recovery of template  $x$  exactly from  $x'$  and  $p$ . The pair of procedures  $SS(\cdot)$  and  $Rec(\cdot)$  represents the secure sketch construction. For different secure sketch algorithms, the  $SS(\cdot)$  and  $Rec(\cdot)$  are different. Based on this framework, Dodis et al. (2004, 2008) proposed several constructions, two of which are adopted in this paper, that are the PinSketch and the improved Juels and Sudan fuzzy vault (IJS fuzzy vault).

The PinSketch construction is based on BCH. In Dodis et al. (2006), the authors have made a standard BCH code based on syndrome encoding and decoding in sublinear time. Here two operations,  $syn(x)$  and  $supp(x)$  are defined, where  $syn(x)$  is the procedure to compute the syndrome of  $x$ , and  $supp(x)$  represents the set of position of  $x$  on which it is nonzero.  $supp(x)$  is also named the support of  $x$ . In Dodis et al. (2006), it states that for a  $[n, k, \delta]$  binary BCH code  $C$  one can compute: (1)  $syn(x)$ , when  $supp(x)$  is given, in time polynomial in  $\delta$ ,  $\log n$ , and  $|supp(x)|$ ; and (2)  $supp(x)$ , when  $syn(x)$  is given, in time polynomial in  $\delta$  and  $\log n$ .

Given a biometric feature vector  $x$ , the PinSketch's sketching procedure  $SS_{ps}(x)$  operates as

$$SS_{ps}(x) = \{s_1, s_2, \dots, s_{2t-1}\} \quad \text{and} \quad s_i = \sum_{a \in \omega} a^i, \quad (10)$$

where  $\omega = supp(x)$  and  $t$  is the tolerance of Hamming distance between templates feature vector and query feature vector. Given a query feature vector  $x'$  and sketch  $SS_{ps}(x)$ , the recovery procedure  $Rec_{ps}(x', SS_{ps}(x))$  operates as

$$Rec_{ps}(x', SS_{ps}(x)) = supp(x') \Delta supp(x), \quad (11)$$

where  $\Delta$  represents set difference operation and  $syn(v) = \{s'_1 - s_1, s'_2 - s_2, \dots, s'_{2t-1} - s_{2t-1}\}$ ,  $s'_i \in SS_{ps}(x')$ . If the Hamming distance of  $dis(x, x') \leq t$ ,  $Rec_{ps}(x', SS_{ps}(x)) = supp(x)$ . More details of PinSketch can be found in Dodis et al. (2006).

The improved Juels and Sudan fuzzy vault is another secure sketch construction. Given an input biometric data set  $x = \{x_1, x_2, \dots, x_s\}$  of size  $s$ ,  $SS_{ijs}(x)$  is defined as

$$SS_{ijs}(x) = \{a_{s-1}, a_{s-2}, \dots, a_{s-t}\}, \quad (12)$$

where  $\{a_{s-1}, a_{s-2}, \dots, a_{s-t}\}$  is the set of coefficients of degree  $s-1$  down to  $s-t$  of polynomial  $p(z) = \prod_{i=1}^s (z - x_i) =$

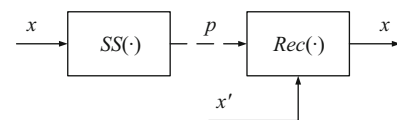


Fig. 5. The flowchart of secure sketch framework.

$x^s + a_{s-1}x^{s-1} + \dots + a_{s-t}x^{s-t} + \dots + 1$ , and  $t$  is the designed tolerance of set difference errors. Given a query biometric data set  $x' = \{x'_1, x'_2, \dots, x'_s\}$ , the recovery procedure  $Rec_{ijs}(x', SS_{ijs}(x))$  is defined as

$$Rec_{ijs}(x', SS_{ijs}(x)) = \{y_1, y_2, \dots, y_s\}, \quad (13)$$

where  $\{y_1, y_2, \dots, y_s\}$  is the roots of polynomial  $p_h - p_l$ , and  $p_h = x^s + a_{s-1}x^{s-1} + \dots + a_{s-t}x^{s-t}$ ,  $p_l$  is a polynomial of degree  $s-t-1$  which holds  $p_l(x'_i) = p_h(x'_i)$  for at least  $s-t/2$  of the  $x'_i$  values.  $[s, s-t, t+1]$  Reed–Solomon decoding algorithm is used here to search such a polynomial. When the set difference between  $x$  and  $x'$  less than or equal to  $t$ , we have  $x = \{y_1, y_2, \dots, y_s\}$ .

The secure sketch techniques provide a reliable way to recover the template data of biometrics. In order to generate a reliable cryptographic key from biometrics, a problem has to be tackled. That is, an efficient key should be randomly and uniformly distributed, while biometrics feature are highly dependant. The universal hash function is a useful tool to reliably extract randomly and uniformly distributed bits string (Carter and Wegman, 1979; Wegman and Carter, 1981). Given a biometric data  $x$ , a cryptographic key can be obtained by

$$key = Ext(x). \quad (14)$$

$Ext(x)$  represents universal hash function operation on  $x$ .

### 3.2. Encoding

The proposed key generation system is based on the fusion of minutiae and modified Biocode, and three sketches are constructed to deal with different kinds of features. The flowchart of encoding stage is shown in Fig. 6. Three kinds of sketch data and the core point with direction will be stored in smart card. The key is generated from the fused features by applying universal hash function.

*Secure sketch construction for minutia set:* A further selection process is taken on the extracted minutia set at encoding stage to make sure that the minimum distance between any two minutiae is greater than  $\delta$ , where  $\delta$  is used to well separate the encoding minutiae and chaff points and is empirically determined from standard deviation of minutiae feature differences. The local quality index (Chen et al., 2005) is used to estimate the quality of each minutia.  $r$  minutiae with high quality index are selected finally.

In the improved Juels and Sudan fuzzy vault (Dodis et al., 2004, 2008) construction, the “lock” set and “unlock” set must have the same size (so it does in the original Juels and Sudan, 2002 fuzzy vault construction). In practice, the template image may not have enough minutiae due to partial observation or lack of minutiae

itself. In this case, the encoding has to stop, but it still has the probability that the templates have large similarity with query images. Because a successful authentication is based on the number of matched minutiae pairs. So, in our implementation, if there are not enough minutiae to be selected, e.g., the number of well-separated minutiae  $N^T < r$ , then  $r - N^T$  randomly generated and well-separated minutia points are added in to form a minutia set  $SM^T = \{m_j^T\}_{j=1}^r$ , where  $SM^T$  is also called encoded minutia set. A chaff point set  $CM = \{m_k\}_{k=1}^s$  is then generated iteratively as described in Nandakumar et al. (2007). The union set  $SM^T \cup CM$  forms our minutiae based sketch  $MS$  that is stored in the smart card.

We add some random minutia points in  $SM^T$  when there are insufficient minutiae to be used, which will eliminate the FTC error rate without producing side effects. The added random minutiae act as real minutiae at encoding stage, but they will be filtered out by query minutia set at decoding stage. The core point's location and its corresponding direction,  $C^T(u_C^T, v_C^T, \theta_C^T)$ , are also detected here using the method described in Section 2.2, and stored in smart card.

*Secure sketch construction for modified Biocode:* At the encoding stage, the Biocode vector is represented as  $B^T$ , where  $B^T \in \{0, 1\}^L$ .  $B^T$  is divided into  $N$  segments, each segment is  $L/N$  bit length, where  $N$  is chosen such that  $L$  is divisible by  $N$ . The resultant Biocode set is represented as  $B^{ST} = \{B_i^{ST}\}_{i=1}^N$ , where  $B_i^{ST}$  is an element in the finite field  $\mathcal{F} = GF(2^{L/N})$ . The sketch data of each Biocode segment are obtained by

$$s_i^B = SS_{ps}(B_i^{ST}), \quad (15)$$

where  $i = 1, 2, \dots, N$ ,  $SS_{ps}$  is defined by Eq. (10), and the tolerance of Hamming distance of each Biocode segment is set to be  $t_1$ . The sketch data and its corresponding sequence numbers are stored in smart card as the Biocode sketch  $BS$ , where  $BS = \{(i, s_i^B) | i = 1, 2, \dots, N\}$ .

*Secure sketch construction for combined feature:* In the combined sketch construction step, the minutia set and Biocode vector are fused together. In  $SM^T$ , the minutia attributes  $u, v$ , and  $\theta$  are quantized as bit strings of length  $l_u, l_v$ , and  $l_\theta$ , respectively, where  $l_u + l_v + l_\theta = L/N$ . By sequentially concatenating the bit string corresponding to  $u, v$ , and  $\theta$ , we convert  $SM^T$  into another set  $SM^T$  in which each element is  $L/N$  bits in length and is an element in the finite field  $\mathcal{F} = GF(2^{L/N})$ . The union set  $MB = SM^T \cup B^{ST}$  with  $r+N$  elements is the combined feature set. We adopt the improved Juels and Sudan fuzzy vault (IJS secure sketch) scheme to construct sketch for  $MB$  (Dodis et al., 2004).

$$CS = SS_{ijs}(MB), \quad (16)$$

where  $SS_{ijs}$  is defined by Eq. (12) and the tolerance of set difference is set to be  $t_2$ .  $CS$  is stored in smart card together with  $MS$  and  $BS$ .

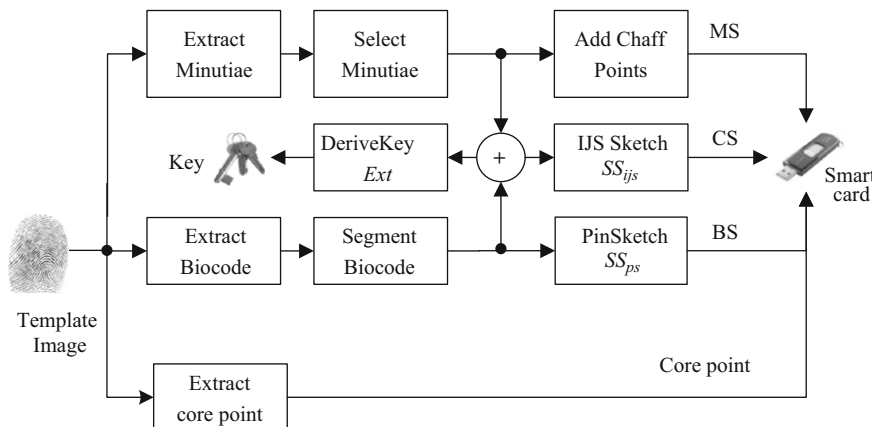


Fig. 6. Flow chart of encoding procedures.

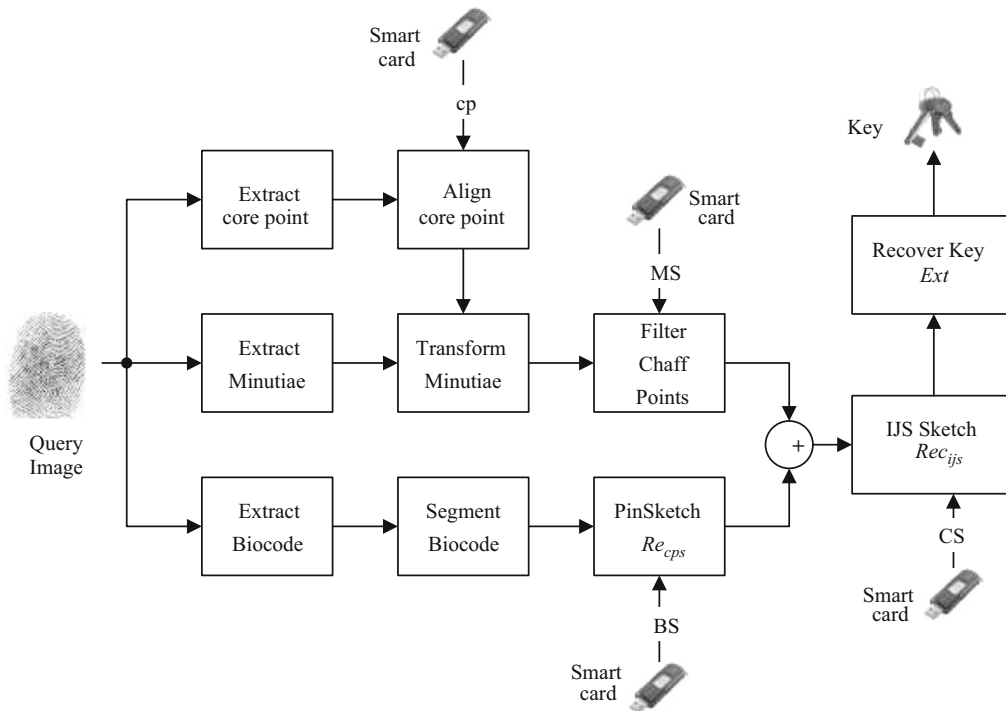


Fig. 7. Flow chart of decoding procedures.

*Key generation by universal hash function:* The last step at encoding stage is to generate the key by

$$key = Ext(MB \parallel MSG), \quad (17)$$

where *Ext* is a universal hash function,  $\parallel$  represents concatenating operation and *MSG* is any plain text which will be stored in smart card too. When *key* is threatened, we can change the content of *MSG* to reissue a new key. Storing *MSG* explicitly in smart card will not leak information about the *key* by the properties of universal hash function.

### 3.3. Decoding

The decoding stage of the key generation system consists of three parts: recovering minutia set, recovering Biocode set, and recovering combined set corresponding to their construction steps, *MS* construction, *BS* construction, and *CS* construction, respectively. Fig. 7 shows the flow chart of decoding procedures.

*Recovering minutia set:* The minutia set recovering procedures operate as follows. Given the query minutia set  $M^Q = \{m_i^Q\}_{i=1}^{N^Q}$ , template core point  $C^T(u_C^T, v_C^T, \theta_C^T)$  and query core point  $C^Q(u_C^Q, v_C^Q, \theta_C^Q)$ , our core point based alignment algorithm described in Section 2.2 is applied on  $M^Q$ , and the aligned query minutia set  $M^{AQ} = \{m_i^{AQ}\}_{i=1}^{N^Q}$  is produced. The next step is to filter out chaff points in *MS* using the aligned query minutiae. We define a distance mapping set  $D = \{d_i\}_{i=1}^{r+s}$  of size  $r+s$ , where  $d_i$  is the minimum distance between  $m_i$  ( $m_i \in MS$ ) and any minutiae in  $M^{AQ}$ . The distance between a template minutia  $m^T(u^T, v^T, \theta^T)$  and an aligned query minutia  $m^Q(u^Q, v^Q, \theta^Q)$  is calculated by

$$D(m^T, m^Q) = d + \beta \Delta \theta, \quad (18)$$

where

$$d = \sqrt{(u^T - u^Q)^2 + (v^T - v^Q)^2},$$

$$\Delta \theta = \min(|\theta^T - \theta^Q|, 360 - |\theta^T - \theta^Q|) - \alpha,$$

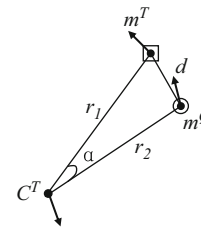


Fig. 8. Minutiae pair distance.

$$\alpha = \arccos \frac{r_1^2 + r_2^2 - d^2}{2r_1 r_2},$$

$$r_1 = \sqrt{(u^T - u_C^T)^2 + (v^T - v_C^T)^2},$$

$$r_2 = \sqrt{(u^Q - u_C^T)^2 + (v^Q - v_C^T)^2}.$$

The distances among template minutia  $m^T$ , query minutia  $m^Q$ , and template core point  $C^T$  are shown in Fig. 8.

The minutia points in *MS* are sorted based on their corresponding distance *D*, and *r* minutiae are sequentially selected starting with the minutia of minimum distance. The selected *r* minutiae form the recovered minutia set  $M^R$ . Due to the noise, the recovered minutia set  $M^R$  is not identical to the encoded minutia set  $SM^T$ . There exists an intersection between  $M^R$  and  $SM^T$ . The size of the intersection indicates the similarity between template fingerprint and query fingerprint.

*Recovering Biocode set:* The Biocode feature vector at the decoding stage is represented as  $B^Q$ .  $B^Q$  is divided into *N* segments, each segment is  $L/N$  bits in length and is an element of the finite field  $\mathcal{F} = GF(2^{L/N})$ . The resultant set is represented as  $B^{SQ} = \{B_i^{SQ}\}_{i=1}^N$ . Restoring the *BS* data from smart card, the

template Biocode can be recovered by

$$B_i^{RT} = Rev_{ps}(B_i^{SQ}, S_i^B), \quad (19)$$

where the function  $Rev_{ps}$  is defined by Eq. (11),  $i = 1, 2, \dots, N$ , and the tolerance of Hamming distance of each segment is  $t_1$  as that at encoding stage. The recovered set is  $B^{RT} = \{B_i^{RT}\}_{i=1}^N$ . According to the recovering property of PinSketch, if the Hamming distance between  $B_i^{ST}$  and  $B_i^{SQ}$  less than or equal to  $t_1$ , then the recovered Biocode segment  $B_i^{RT}$  is identical to  $B_i^{ST}$ , otherwise, the value of  $B_i^{ST}$  does not make any sense.

**Recovering combined set:** For each minutia  $m_i(u_i, v_i, \theta_i)$  in  $M^R$ ,  $u_i, v_i$ , and  $\theta_i$  are quantized into bit strings of length  $l_u, l_v$ , and  $l_\theta$ , respectively, where  $l_u + l_v + l_\theta = L/N$ . By sequentially concatenating bit strings corresponding to  $u, v$ , and  $\theta$ , we can convert  $M^R$  into another set  $M^R$  in which each element is  $L/N$  bits in length and is an element in the finite field  $\mathcal{F} = GF(2^{L/N})$ .  $M^R$  and Biocode set  $B^{RT}$  are united together to obtain  $MB^R = M^R \cup B^{RT}$ . Let  $t_m$  be the set difference between  $M^T$  and  $M^R$ , and  $t_b$  be the set difference between  $B^{ST}$  and  $B^{RT}$ , then the set difference between  $MB$  and  $MB^R$  is  $t_m + t_b$ . We adopt the recovery procedure of IJS fuzzy vault to recover the encoded combining set by

$$MB' = Rec_{ijs}(MB^R, CS), \quad (20)$$

where  $MB'$  is the recovered combined feature, the function  $Rec_{ijs}$  is defined by Eq. (13), and the tolerance of set difference in  $Rec_{ijs}$  is set to be  $t_2$ . If  $t_m + t_b > t_2$ ,  $Rec_{ijs}$  will return "fail", otherwise,  $MB'$  is equal to  $MB$ . When  $MB$  is identically recovered, we can restore the key by

$$key = Ext(MB' \parallel MSG), \quad (21)$$

where  $Ext$  is the same universal hash function as that used at encoding stage.

## 4. Experimental results

### 4.1. Experiment setups

The proposed key generation system is evaluated on FVC2002-DB1 and FVC2002-DB2 fingerprint databases (Maio et al., 2002). These two databases are public-domain databases and have relatively high image quality. Both database contain 100 fingers (objects), each finger have eight impressions available. Only impressions 1 and 2 are used in our key generation experiments because these two impressions are acquired in the same session and have relatively small translations and rotations to each other. In a biometric cryptosystem, it is reasonable to assume that the

users are cooperative and are willing to provide good quality biometric data in order to retrieve their cryptographic keys (Nandakumar et al., 2007).

The performance indices used for evaluating the proposed system are genuine accept rate (GAR) and false accept rate (FAR). The GAR is defined as the ratio of successful genuine attempts number to total genuine attempts number. In our key generation experiment, impression 1 is used as template images and impression 2 as query images, each object has one genuine attempt, so the total number of genuine attempts is 100 for both databases. The FAR is the percentage of attempts made by imposters that resulted in successful key recovery. The Imposter attempts were simulated between any two identities, the template and query images are randomly selected from impression 1 or 2. The number of imposter attempts is 4950.

At encoding stage, minutia features  $SM^T$  and Biocode features  $B^{ST}$  are fused together as combined features  $MB$  by the equation  $MB = SM^T \cup B^{ST}$ . By controlling the selection of  $SM^T$  and  $B^{ST}$ , the system will become one that only based on minutiae features or Biocode features. The system that based on minutiae, Biocode, and fusion of minutiae and Biocode are denoted as Minu-System, Bio-System, and Fusion-System, respectively.

### 4.2. Single feature based system

When  $B^{ST}$  is set to empty, the system is a fuzzy vault system, or Minu-System. The minimum distance  $\delta$  between any two points in  $MS$  is set to be 25 for FVC2002-DB1 and 30 for FVC2002-DB2. Two hundred chaff points are added in minutia based sketch  $MS$  considering of both security and storage. The size of encoded minutia set  $SM^T$  is chosen to be 20 for DB1 and 25 for DB2. When there are not enough real minutia points of the template to be chosen, some random minutia points are generated and treated as real minutia points. The Minu-System performance is shown in Fig. 9 in which  $t_2$  indicates the set differences between  $MB$  and  $MB^R$ , left and right vertical axis indicate GAR and FAR value corresponding to GAR curves and FAR curves, respectively. From the figure, the performance on DB1 is not as good as that on DB2, and set difference  $t_2$  on DB1 is higher than that on DB2, this is because the images in DB2 contain more effective areas of fingertips.

In the Bio-System,  $SM^T$  is set to be empty, and the system is solely based on Biocode. In this system, the 1024-bit Biocode feature vector is divided into 32 segments, each segment is 32 bits length, the tolerance  $t_1$  used in PinSketch procedures is set to be 10, 11, 12, 13, and 14, and the performance is shown in Fig. 10(a) and (b). On Fig. 10(a) and (b), the curves cross over each other. No

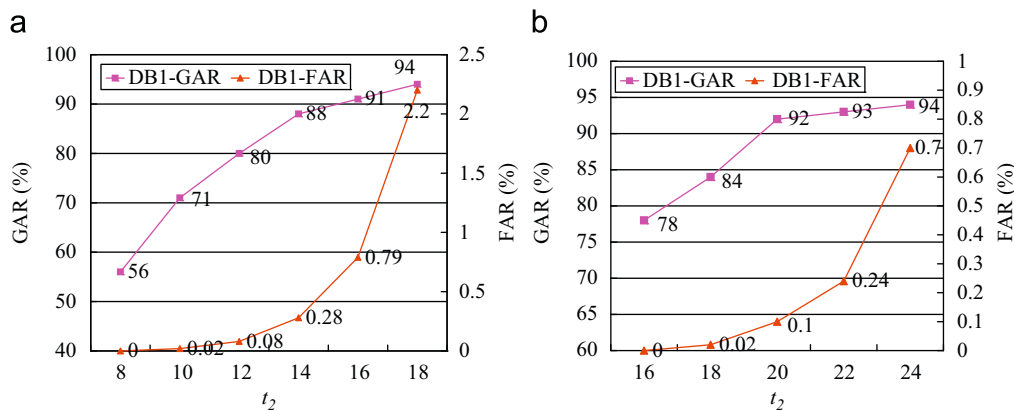


Fig. 9. Performance of Minu-System on (a) FVC2002-DB1 and (b) FVC2002-DB2.

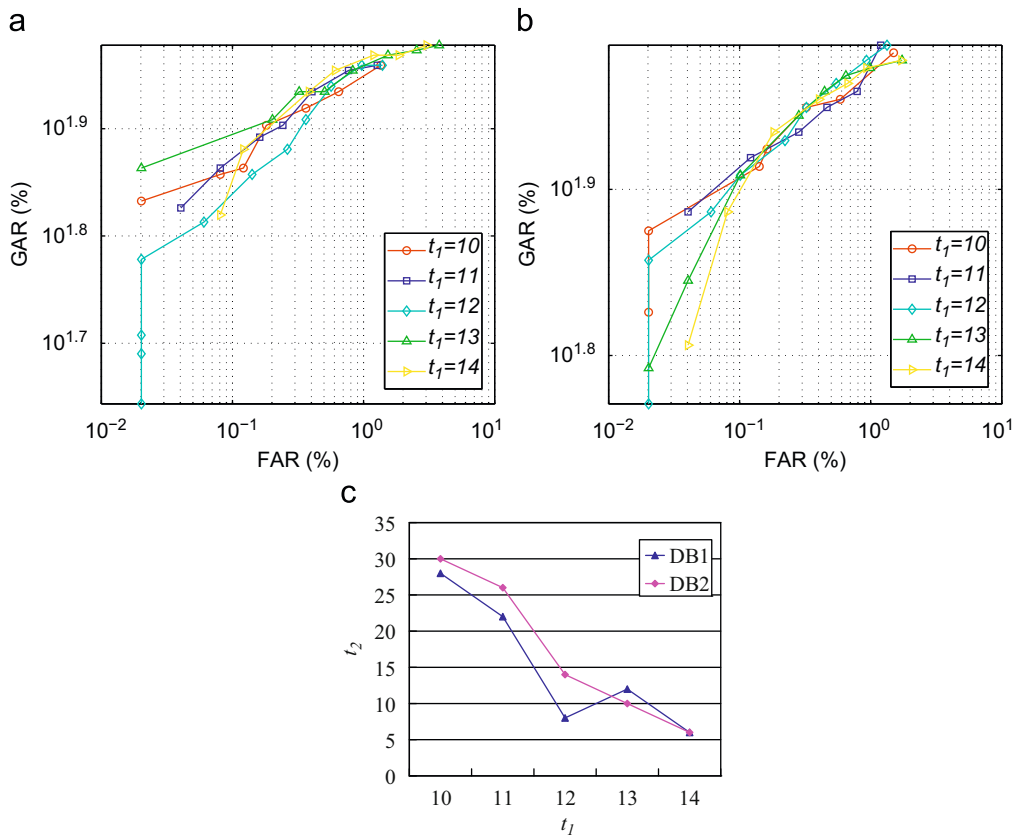


Fig. 10. ROC curves of Bio-System on (a) FVC2002-DB1; (b) FVC2002-DB2 by setting  $t_1$  to different values; (c) the relation of  $t_1$  and  $t_2$  when FAR = 0.

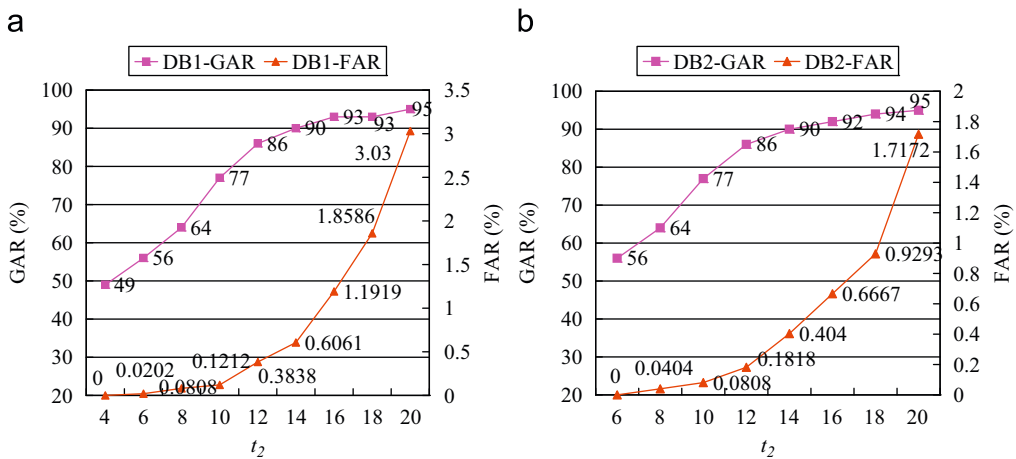


Fig. 11. Performance of Bio-System on (a) FVC2002-DB1; (b) FVC2002-DB2 when  $t_1 = 14$ .

one outperforms all the others, hence the Bio-System performance has no strong relations to  $t_1$ . But when  $t_1$  is too small, the tolerance of IJS fuzzy vault  $t_2$  have to set to a big value to seek a good balance between FAR and FRR. Fig. 10(b) shows the relations between  $t_1$  and  $t_2$  when the Bio-System's FAR is 0. The higher tolerance of  $t_2$  means more block errors happen in Biocode. This is because lower  $t_1$  means that less errors can be tolerated in one block which will produce more block errors. In our experiments,  $t_1$  is set to be 14 because it gives a good balance between the minutiae errors and Biocode block errors. Fig. 11 shows the experimental results on FVC2002-DB1 and FVC2002-DB2 when  $t_1 = 14$ .

### 4.3. Multiple feature based system

The Fusion-System takes account for both minutiae and Biocode features. The parameters for DB1 is set as  $r = 20$ ,  $\delta = 25$  and  $t_1 = 14$ , and for DB2  $r = 25$ ,  $\delta = 30$  and  $t_1 = 14$ . The performance results are shown in Figs. 12 and 13. From the figure, we can see that though Minu-System and Bio-System both have a low GAR, the Fusion-System outperforms each single feature based system greatly. On FVC2002-DB1, when FAR = 0, the GAR is 56% for Minu-System and 49% for Bio-System, whereas in Fusion-System, the GAR is improved to 85%. On FVC2002-DB2, there is similar conclusion. These results demonstrate that

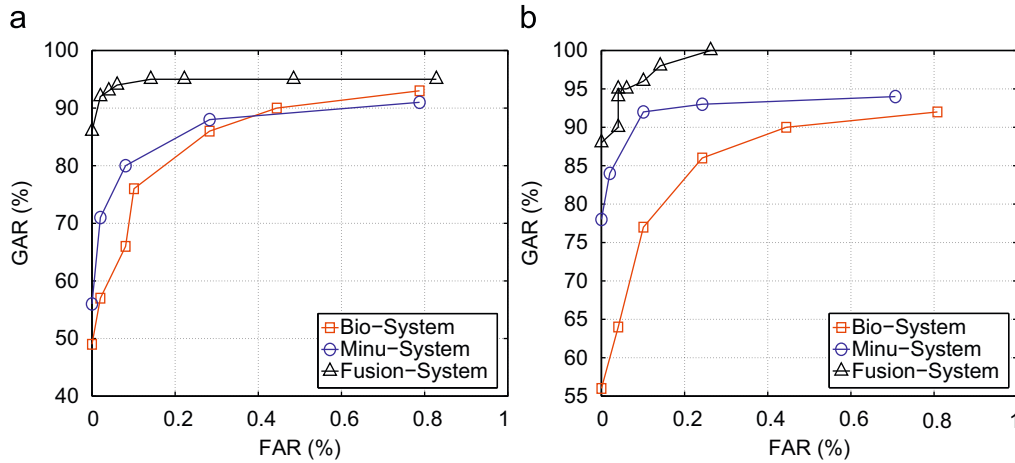


Fig. 12. ROC curves of Bio-System, Minu-System, and Fusion-System on (a) FVC2002-DB1 and (b) FVC2002-DB2.

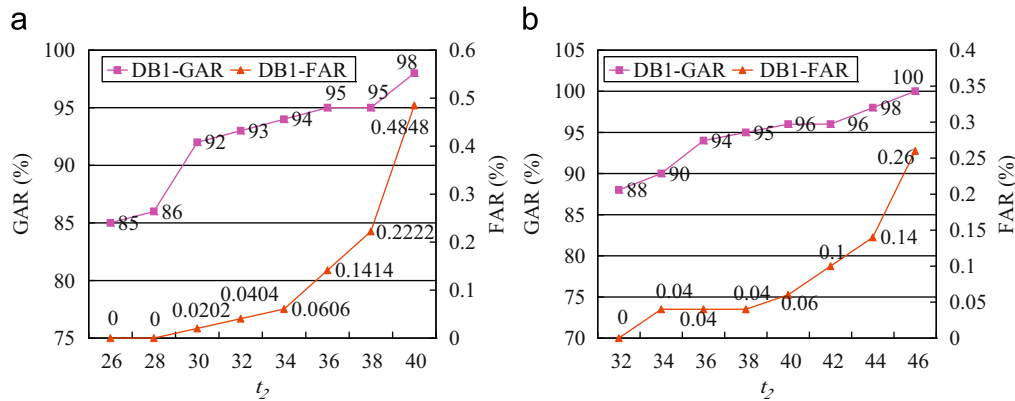


Fig. 13. GARs and FARs of the Fusion-System on (a) FVC2002-DB1 and (b) FVC2002-DB2.

Table 1  
EERs of three systems on DB1 and DB2 (%).

	DB1	DB2
Bio-System	3.52	2.92
Minu-System	5.08	3.01
Fusion-System	3.81	1.07

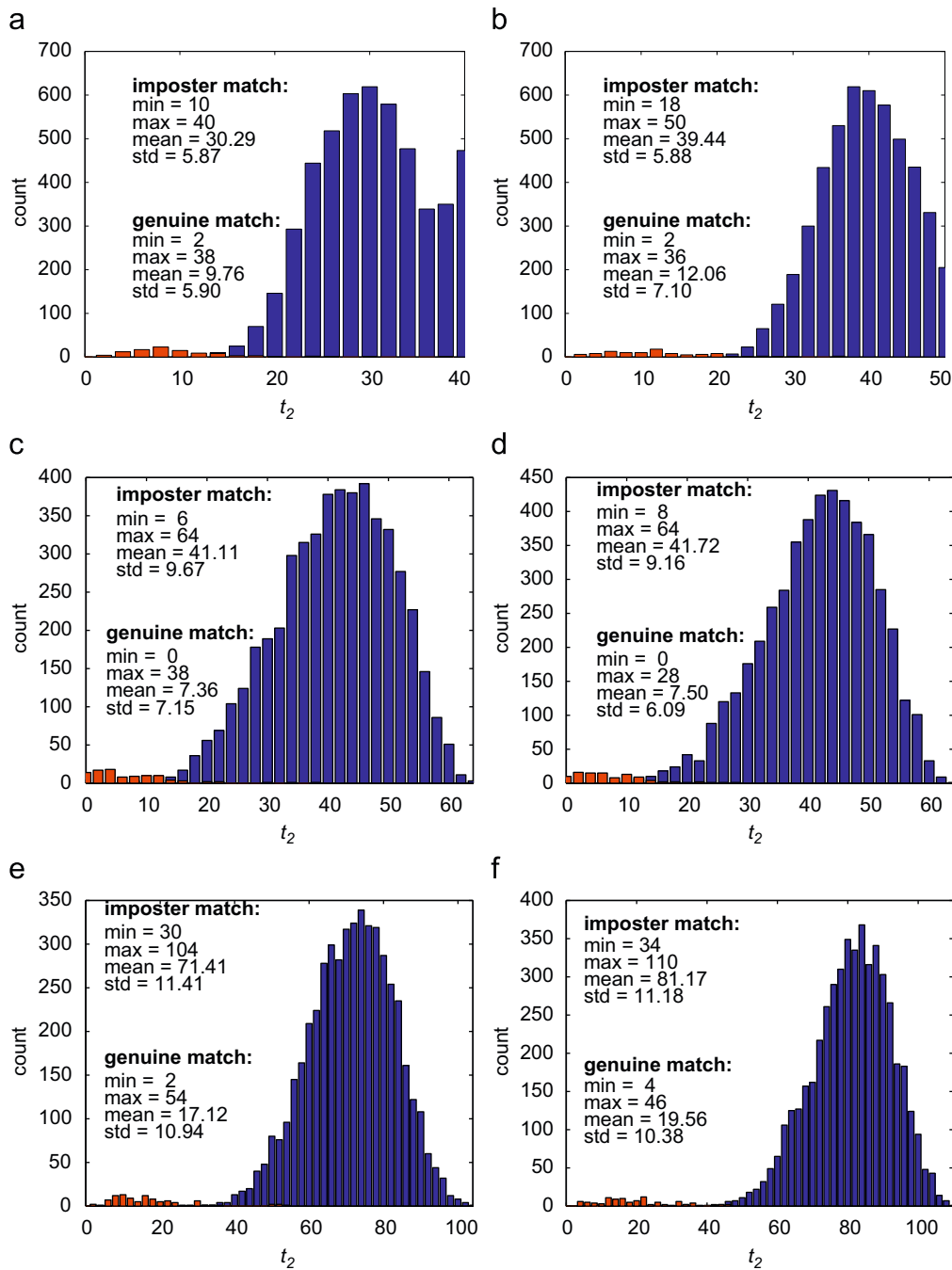
minutiae and Biocode represent two kinds of different characteristics of fingerprint images and have much information to complement each other.

Table 1 shows the equal error rates (EER) of Bio-System, Minu-System and Fusion-System on DB1 and DB2. We see from the table that both on DB1 and DB2, Bio-System gets a lower EER than Minu-System (even lower than Fusion-System on DB1), but when FAR at a low level, say FAR less than 0.4%, Minu-System outperforms Bio-System (see Fig. 12). Moreover, when FAR at a low level, Fusion-System outperforms both Bio-System and Minu-System drastically (see Fig. 12). In a biometric cryptosystem, false accept is usually more harmful than false reject, so it is important to keep the system running at a low FAR level while still getting a high GAR, which is a more practical situation.

Fig. 14 shows the histograms of distribution of set difference  $t_2$  in all three systems on FVC2002-DB1 and FVC2002-DB2 databases. From the figure, we get the conclusion that the peaks of genuine match and imposter match in the Fusion-System is getting more apart from each other compared with Minu-System

or Bio-System. On FVC2002-DB1, the peak distance between imposter match and genuine match of Minu-System and Bio-System is 20.53 and 33.75, respectively, while in the Fusion-System, the peak distance is 54.29. Meanwhile, the genuine standard deviation of  $t_2$  in Fusion-System is enlarged by around 4.41 and the imposter standard deviation is enlarged by around 3.6. The same conclusion can be get from FVC2002-DB2. By fusing minutiae and Biocode, the genuine attempts and imposter attempts become more distinguishable.

There are eight kinds of situations about the decoding results for each object in three system, Bio-System, Minu-System, and Fusion-System. Table 2 summarizes the 100 genuine attempts of these situations. In this table, each row represents one situation, in which “×” represents failure authentication and “√” represents successful authentication, e.g., row 2 represents the number of objects which result in success in Bio-System, failure in Minu-System, and success in Fusion-System. All three systems, Bio-System, Minu-System, and Fusion-System, work in the state of FAR = 0. From the table, we can see that there is a large number of objects which failure in Bio-System or Minu-System but success in Fusion-System. The situation which failure in both Bio-System and Minu-System but success in Fusion-System also exist (14 cases in DB1 and four cases in DB2). When Fusion-System failure, the situation of both Bio-System and Minu-System failure have the most contribution, 92.86% in DB1 and 66.7% in DB2, the rest of three situations have very low shares. The foreground regions of fingerprint in DB2 is larger than in DB1, and more minutiae can be extracted in DB2, so in situations 1 and 3, in



**Fig. 14.** Distribution of set difference  $t_2$ , from left to right, columns 1 and 2 represent FVC2002-DB1 and FVC2002-DB2, respectively; from top to down, rows 1,2,and 3 represent Minu-System, Bio-System and Fusion-System, respectively.

**Table 2**

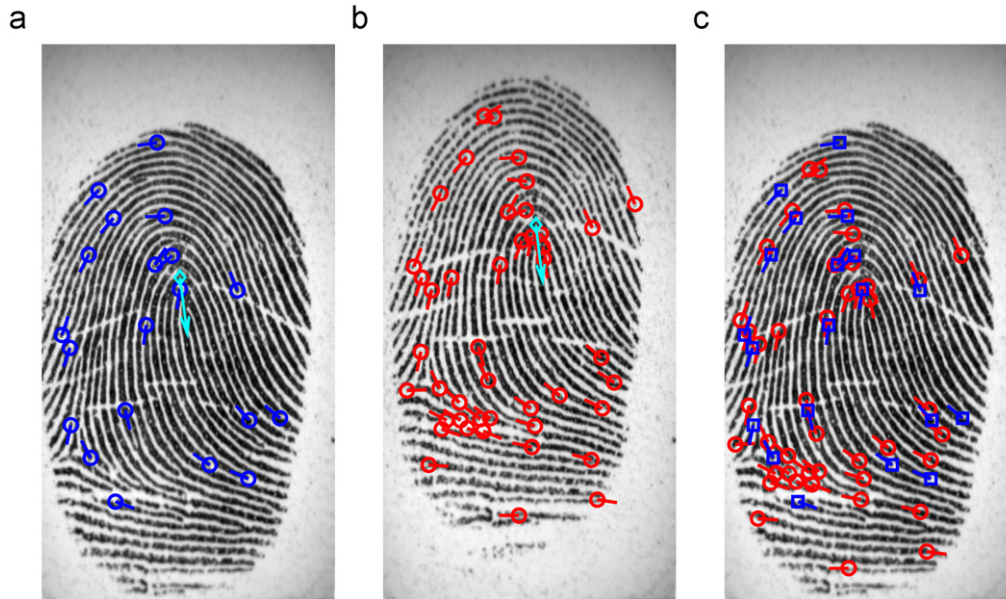
Genuine imposter distribution when all three systems working in the state of FAR = 0 on FVC2002-DB1 and FVC2002-DB2.

Situation	Bio-System	Minu-System	Fusion-System	DB1	DB2
1	✓	✓	✓	38	46
2	✓	×	✓	19	9
3	×	✓	✓	17	29
4	×	×	✓	12	
5	✓	✓	×	0	0
6	✓	×	×	0	1
7	×	✓	×	1	3
8	×	×	×	13	8

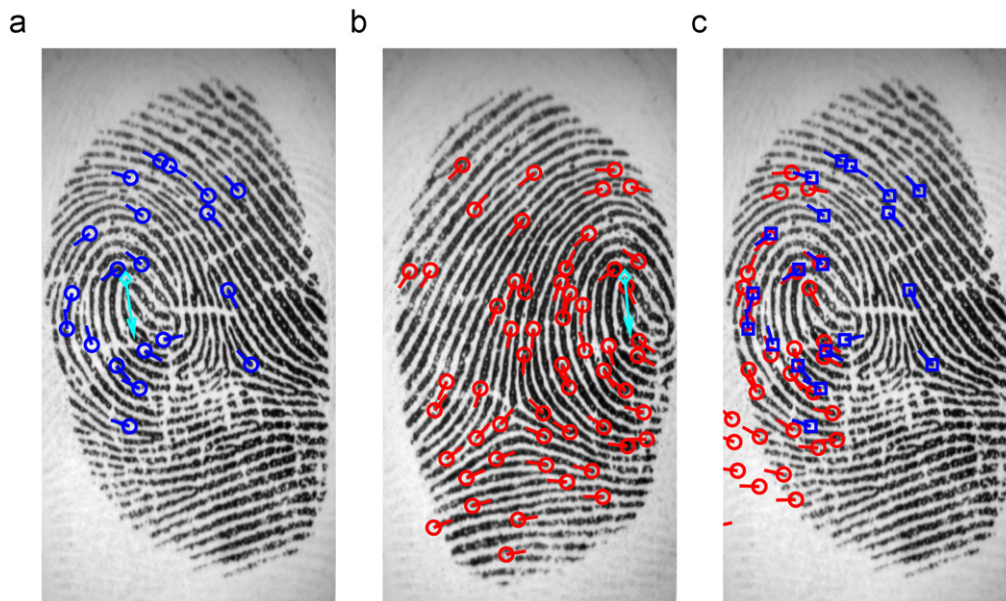
The IJS fuzzy vault tolerance  $t_2$  for Bio-System, Minu-System, and Fusion-System is set to be 6, 8, and 28 on DB1, and 6, 16, and 32 on DB2.

which Minu-System success, the number of objects in DB2 which success in Fusion-System is larger than that in DB1. Figs. 15–17 show three examples of situations 2, 3, and 8, respectively.

In our system, we add some random minutiae to  $SM^T$  when there are not enough minutiae to be used at the encoding stage. These random minutiae act like real minutiae but will be filter out at decoding stage with large probability. In this way, we keep the potential objects who do not have sufficient minutiae being able to register in the system. Table 3 presents some numbers about these objects, in which both Minu-system and Fusion-System work in the state of FAR = 0. The table tell us that large parts of the template fingerprints which have insufficient minutiae by adding random minutiae will get a successful authentication at decoding stage.



**Fig. 15.** An example of successful key recovery in Fusion-System while failure in Bio-System and success in Minu-System; (a) template image and selected 25 minutiae; (b) query image and minutiae; (c) aligned minutiae; the set difference  $t_2$  in Bio-System, Minu-System, and Fusion-System is 10 ( $> 6$ ), 4 ( $< 16$ ) and 14 ( $< 32$ ), respectively. Based on Biocode it is unable to recover the key, but when minutiae get involved, the authentication is successful. The marker “ $\diamond$ ” with an arrow “ $\searrow$ ” indicates the location and direction of core point.



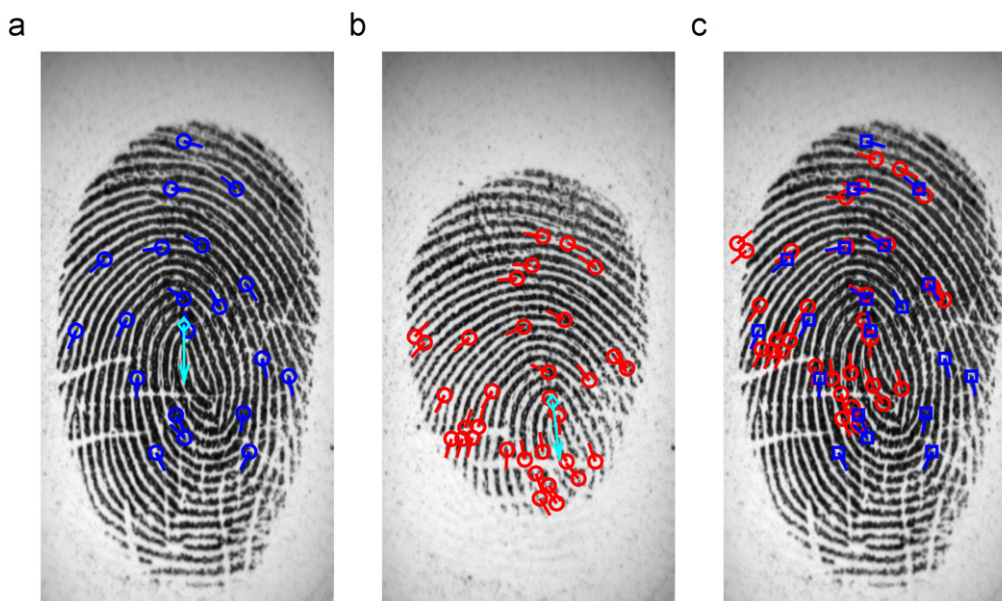
**Fig. 16.** An example of successful key recovery in Fusion-System while both failure in Bio-System and Minu-System; (a) template image and selected 25 minutiae; (b) query image and minutiae; (c) aligned minutiae; the set difference  $t_2$  in Bio-System, Minu-System, and Fusion-System is 12 ( $> 6$ ), 20 ( $> 16$ ) and 32 ( $= 32$ ), respectively. Due to partial observation and poor quality, both the Bio-System, and Minu-System fail to recover the key, but it successes in Fusion-System. The marker “ $\diamond$ ” with an arrow “ $\searrow$ ” indicates the location and direction of core point.

Our fusion-system’s performance is also compared with the results of fuzzy vault system proposed in Nandakumar et al. (2007) and Nagar et al. (2008) on FVC2002-DB2. The comparison results are shown in Table 4. When FAR is on the same or similar levels, our algorithm have higher GAR except the case of FAR = 0, and there is no failure to capture (FTC) errors in our system, which makes our system’s GAR can achieve 100% with a relatively low FAR. Both in Nandakumar et al. (2007) and Nagar et al. (2008), the FTC rate is 2%. When FAR = 0, Nagar’s method has a higher GAR than that in the proposed method, the main reason is that in

Nagar et al. (2008), minutiae’s coordinates are all secured with their corresponding descriptors, and the chaff points are combined with random descriptors. Although their method can efficiently reduce FAR, the GAR is not improved.

#### 4.4. Alignment performance

The alignment accuracy have a large influence on the performance of the whole system. To evaluate the proposed core



**Fig. 17.** An example of failure key recovery in all three systems; (a) template image and selected 25 minutiae; (b) query image and minutiae; (c) aligned minutiae; The set difference  $t_2$  in Bio-System, Minu-System, and Fusion-System is 10 (> 6), 24 (> 16) and 34 (> 32), respectively. Query fingerprint has very poor quality which results in all three systems fail to recover the key. The marker “ $\diamond$ ” with an arrow “ $\searrow$ ” indicates the location and direction of core point.

**Table 3**

Statistical number about how many objects succeed to be authenticated in Minu-System and Fusion-System when the real minutiae are insufficient.

Database	DB1	DB2
$Num_t$	7	6
$Num_m$	3	4
$Num_f$	5	6

( $Num_t$  indicates the total number of objects with insufficient minutiae;  $Num_m$  indicates the number of objects which have insufficient minutiae success in Minu-System;  $Num_f$  indicates the number of objects which have insufficient minutiae success in Fusion-System.)

**Table 4**

Performance comparison with Nandakumar et al. (2007) and Nagar et al. (2008) on FVC2002-DB2.

Algorithm	FTC	GAR	FAR	GAR	FAR	GAR	FAR	GAR	FAR
Nandakumar et al. (2007)	2	95	0.7	91	0.13	91	0.01	86	0
Nagar et al. (2008)	2	–	–	93	0.1	93	0.01	93	0
Proposed	0	100	0.262	96	0.101	95	0.04	88	0

“–” means not reported in their papers (%).

direction algorithm, FVC2002 DB2 impressions 1 and 2 are selected for comparison. We manually align impressions 1 and 2 of each subject. The manual alignment procedure is performed by selecting some control points between impressions 1 and 2. These control points is used by Goshtasby (1986) to estimate the rigid transformation parameters. Based on these transformation parameters, all the minutiae of impression 2 are aligned with impression 1. Then the paired minutiae are determined by bounding box (Jain et al., 1997). Finally, the paired minutiae are checked and revised manually. By the above operations, we create 3751 minutia pair samples. Complex filtering method (Nilsson and Bigun, 2003) is selected for comparison. The 3751 pairs of minutiae are aligned by manually aligning, complex filtering and

**Table 5**

Alignment accuracy comparison with manual alignment and complex filtering (Nilsson and Bigun, 2003) on FVC2002 DB2.

	$mean_{ \Delta u }$	$mean_{ \Delta v }$	$mean_{ \Delta \theta }$	$std_{ \Delta u }$	$std_{ \Delta v }$	$std_{ \Delta \theta }$
Manual alignment	3.3268	3.8299	4.6480	4.2105	4.6011	6.2832
Nilsson and Bigun (2003)	10.1057	9.5648	6.1417	15.8892	17.8892	6.7402
Proposed method	5.7123	6.8744	5.3186	8.6640	11.7457	6.9020

the proposed method, respectively. The mean absolute differences  $mean_{|\Delta u|}$ ,  $mean_{|\Delta v|}$  and  $mean_{|\Delta \theta|}$  and the corresponding standard deviations  $std_{|\Delta u|}$ ,  $std_{|\Delta v|}$  and  $std_{|\Delta \theta|}$  are used for evaluation criteria (see Table 5).

The experimental results show that our core based alignment algorithm outperforms complex filtering. Our algorithm has a very low average alignment errors and standard deviations. We also see that the standard deviations of  $|\Delta \theta|$  obtained by manual alignment, complex filtering and the proposed method do not vary too much, which means the minutia direction is more stable than location coordinates under rigid transformation of fingerprint image.

#### 4.5. Performance of modified Biocode

The modified Biocode features are tested on a subset of FVC2002 DB2. Six impressions with large core area of each subject are selected. The genuine attempts are simulated between any two of the six selected impressions, which yields to 1500 genuine attempts. The imposter attempts are simulated between any two objects, impression 1 is used. The number of imposter attempts are 4950. The similarity of two Biocode is defined as the number of bit differences.

Seven matchers are tested in this experiment, they are:

- (1) Base128: The base BioHashing with an output feature vector of length 128.

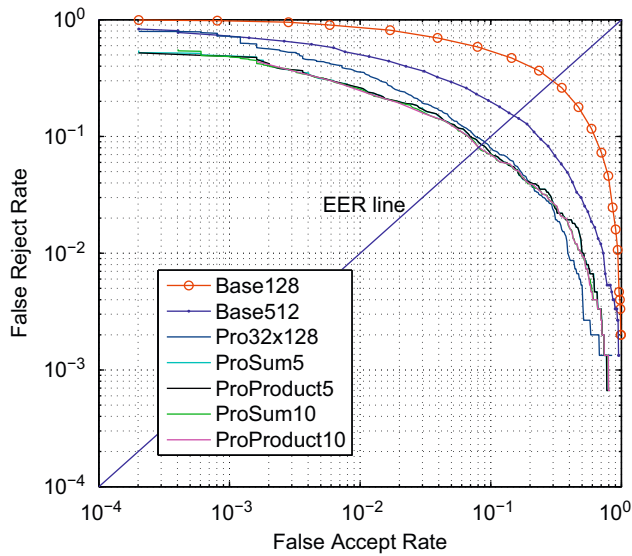


Fig. 18. The ROCs of based Biocode and the modified Biocode algorithm.

Table 6  
EER comparison between the modified Biocode and base Biocode (Jin et al., 2004a) (%).

	Base128	Base512	Pro32x128	ProSum5	ProProduct5	ProSum10	ProProduct10
EER	30.55	15.3	8.99	8.49	8.51	8.07	8.08

- (2) Base512: The base BioHashing with an output feature vector of length 512.
- (3) Pro32 × 128: The modified BioHashing with random matrix  $R$  having size  $32 \times 128$ .
- (4) ProSum5: An ensemble of five matchers fused by sum rule.
- (5) ProProduct5: An ensemble of five matchers fused by product rule.
- (6) ProSum10: An ensemble of 10 matchers fused by sum rule.
- (7) ProProduct10: An ensemble of 10 matchers fused by product rule.

The above four ensembles of matcher are obtained by different random matrix  $R$ . All the random matrix size of  $R$  are set to 32. Sum rule and product rule are used to fuse the matchers. More details about ensemble of matcher we refer to (Nanni and Lumini, 2006a).

The experimental results of modified Biocode are shown in Fig. 18. The results show that our modified Biocode outperforms the conventional Biocode. The EERs are shown in Table 6. Ensemble of matchers outperform Pro32 × 128. More matchers fused can improve the matching performance.

### 5. Security analysis

If the smart card is safe, the security level of the system is equivalent to guess the feature of fingerprint image, which is computationally infeasible. The key is generated by universal hash function, and the key size depends on certain hash function, e.g., the output of SHA-1 is 160 bits in length. We suppose that an attacker has the smart card and read the sketch data  $MS$ ,  $BS$ , and  $CS$ , and also the core point information. We compute the system entropy given the sketches in different scenarios.

$CS$  is constructed by IJS fuzzy vault (Dodis et al., 2004). By the security property of IJS fuzzy vault, the combined features  $MB$  can

be recovered by the adversary who observes  $CS$  with probability no greater than  $2^{-\tilde{m}_1}$ , where  $\tilde{m}_1$  is the entropy of  $MB$  given  $CS$ . In our implementation of IJS fuzzy vault,  $\tilde{m}_1 = N(r+L/N) - t_2 \log(2^N - 1)$ , where  $N(r+L/N)$  indicates the total entropy of  $MB$ , and  $t_2 \log(2^N - 1)$  is the entropy loss of  $MB$  given  $CS$ . The security level is about 768 bits on FVC2002-DB1 and 800 bits on FVC2002-DB2. From  $CS$ , the adversary can get nothing about the key.

The adversary may also guess the combined features  $MB$  from  $MS$  and  $BS$ . We consider the brute force attack here. The attacker do not need to guess all elements in  $MB$  to recover the key, only  $r+L/N-t_2/2$  points is enough, because the correctness property of IJS fuzzy vault guarantee recovery of  $MB$  when there are enough real points at hand. There are  $r$  elements in  $MS$  which are also contained in  $MB$ . The probability for an attacker to guess  $n_1 (1 \leq n_1 \leq r)$  points of  $MB$  given  $MS$  is

$$p_{MS}(n_1) = \binom{r}{n_1} / \binom{r+s}{n_1}. \tag{22}$$

The  $BS$  sketch is constructed by PinSketch (Dodis et al., 2004). From one segment of sketch  $s_i^B$ , the attacker can guess  $B_i^{ST}$ , contained also in  $MB$ , with probability no great than  $p_{ps} = 2^{-\tilde{m}_2}$ , where  $\tilde{m}_2$  is the entropy of  $B_i^{ST}$  given  $s_i^B$ . PinSketch is designed for large universe, while in a small universe, it is reduced to a code-offset construction. In this case, the entropy  $\tilde{m}_2$  of  $B_i^{ST}$  given  $s_i^B$  is  $\log(2^{L/N} / (L/N/t_1))$  by sphere-packing bound (MacWilliams and Sloane, 2003). The probability for an attacker to guess  $n_2 (1 \leq n_2 \leq L/N)$  points of  $MB$  given  $BS$  is

$$p_{BS}(n_2) = (p_{ps})^{n_2} = 2^{-n_2 \tilde{m}_2}. \tag{23}$$

So, the probability for an attacker to guess  $r+L/N-t_2/2$  points of  $MB$  given  $MS$  and  $BS$  is

$$p = \sum_{n_1, n_2} p_{MS}(n_1) p_{BS}(n_2), \tag{24}$$

where

$$\begin{cases} n_1 + n_2 = r + \frac{L}{N} - \frac{t_2}{2}, \\ 1 \leq n_1 \leq r, \\ 1 \leq n_2 \leq \frac{L}{N}. \end{cases}$$

In this scenarios, the entropy of the system is

$$H = -\log p. \tag{25}$$

In the proposed key generation system, the parameters is set  $L = 1024$ ,  $N = 32$ ,  $s = 200$ ,  $t_1 = 14$ ,  $r = 20$  and  $t_2 = 28$  for FVC2002 DB1 and  $L = 1024$ ,  $N = 32$ ,  $s = 200$ ,  $t_1 = 14$ ,  $r = 25$  and  $t_2 = 32$  for FVC2002 DB2. The system entropy  $H$  is about 116 bits on FVC2002 DB1 and 125 bits on FVC2002 DB2. The actual security level may be a little lower than that because Biocode is not ideally randomly distributed.

### 6. Conclusion and future work

Reliably extracting a cryptographic key from fingerprint image is still a challenging problem due to large intra-user variation and limited matching manner. Single feature based key generation system is hard to obtain high performance. Multi-feature based method gives a new solution to achieve higher GAR and lower FAR.

In this paper, we propose a fingerprint key generation system under the framework of fuzzy extractor by fusing minutiae and modified Biocode. Three sketches are constructed for minutiae features, Biocode features and the combined features to secure template and handle noises. To align the template and query image at decoding stage, we propose a novel algorithm to extract

stable direction of core point to accomplish the aligning task. Experimental results show that our system can efficiently improve the performance compared with the system based only on minutiae or modified Biocode.

The performance of the proposed fusion scheme in the key generation system can be further improved by fusing more features, e.g., fingercode, orientation field, of course the fusion strategy may need some changes. All features of fingerprint should be changed into the features that can be handled by some known secure sketch constructions.

## Acknowledgments

This paper is supported by the Project for Cheung Kong Scholars and Innovative Research Team in University (PCSIRT) under Grant No. IRT0645, the Chair Professors of Cheung Kong Scholars Program of Ministry of Education of China, 863 program under Grant no. 2008AA01Z411, the National Natural Science Foundation of China under Grant no. 60621001, 60875018, CAS Hundred Talents Program, the Key Program of NSFC-Guangdong Union Foundation under Grant no. U0835004, Beijing Natural Science Fund under Grant no. 09D0532.

## References

- Bazen A, Gerez S. Systematic methods for the computation of the directional fields and singular points of fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2002;24(7):905–19.
- Blum L, Blum M, Shub M. A simple unpredictable pseudo-random number generator. *SIAM Journal on Computing (Print)* 1986;15(2):364–83.
- Bose R, Ray-Chaudhuri D. On a class of error-correcting binary group codes. *Information and Control* 1960;3(1):68–79.
- Bringer J, Chabanne H, Cohen G, Kindarji B, Zemor G. Theoretical and practical boundaries of binary secure sketches. *IEEE Transactions on Information Forensics and Security* 2008;3(4):673–83.
- Carter J, Wegman M. Universal classes of hash functions. *Journal of Computer and System Sciences* 1979;18:143–54.
- Chen Y, Dass SC, Jain AK. Fingerprint quality indices for predicting authentication performance. In: *Proceedings of AVBPA, Lecture notes in computer science*, vol. 3546. Rye Brook, USA: Springer; 2005. p. 160–70.
- Chikkerur S, Cartwright A, Govindaraju V. Fingerprint enhancement using STFT analysis. *Pattern Recognition* 2007;40(1):198–211.
- Chung Y, Moon D, Lee S, Jung S, Kim T, Ahn D. Automatic alignment of fingerprint features for fuzzy fingerprint vault. In: *Lecture notes in computer science*, vol. 3822, 2005. p. 358.
- Clancy TC, Kiyavash N, Lin DJ. Secure smartcardbased fingerprint authentication. In: *Proceedings of the 2003 ACM SIGMM workshop on biometrics methods and applications*. New York, NY, USA: ACM. p. 45–52.
- Dass SC, Jain AK. Fingerprint classification using orientation field flow curves. In: *Proceedings of the Indian conference on computer vision, graphics and image processing*. p. 650–5.
- Dodis Y, Ostrovsky R, Reyzin L, Smith A. Syndrome encoding and decoding of BCH codes in sublinear time. Available at: <<https://www.cs.bu.edu/faculty/reyzin/code/bch-excerpt.pdf>>; 2006.
- Dodis Y, Ostrovsky R, Reyzin L, Smith A. Fuzzy extractors: how to generate strong keys from biometrics and other noisy data. *SIAM Journal on Computing* 2008;38(1):97–139.
- Dodis Y, Reyzin L, Smith A. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In: *Advances in cryptology—EUROCRYPT '2004. Lecture notes in computer science*. Berlin, Germany: Springer.
- Fan L, Wang S, Wang H, Guo T. Singular points detection based on zero-pole model in fingerprint images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2008;30(6):929–40.
- Giuliano E, Paita O, Stringa L. Electronic Character-Reading System, 1977. US Patent 4,047,152.
- Goshtasby A. Piecewise linear mapping functions for image registration. *Pattern Recognition* 1986;19(6):459–66.
- Hao F, Anderson R, Daugman J. Combining crypto with biometrics effectively. *IEEE Transactions on Computers* 2006. p. 1081–8.
- Hocquenghem A. Codes Correcteurs D'erreurs. *Chiffres* 1959;2(2):147–56.
- Huckemann S, Hotz T, Munk A. Global models for the orientation field of fingerprints: an approach based on quadratic differentials. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2008;30(9):1507–19.
- Jain A, Hong L, Bolle R. On-line fingerprint verification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1997;19(4):302–14.
- Jain A, Prabhakar S, Hong L, Pankanti S. Filterbank-based fingerprint matching. *IEEE Transactions on Image Processing* 2000;9(5):846–59.
- Ji L, Yi Z. Fingerprint orientation field estimation using ridge projection. *Pattern Recognition* 2008;41(5):1508–20.
- Jin ATB, Ling DNC, Gohb A. Biohashing: two factor authentication featuring fingerprint data and tokenised random number. *Pattern Recognition* 2004a;37(11):2245–55.
- Jin ATB, Ling DNC, Song OT. An efficient fingerprint verification system using integrated wavelet and fourier–mellin invariant transform. *Image and Vision Computing* 2004b;22(6):503–13.
- Juels A, Sudan M. A fuzzy vault scheme. In: *Proceedings of 2002 IEEE international symposium on information theory*.
- Juels A, Sudan M. A fuzzy vault scheme. *Designs, Codes and Cryptography* 2006; 38(2):237–57.
- Juels A, Wattenberg M. A fuzzy commitment scheme. In: *Proceedings of the 6th ACM conference on computer and communications security*. New York, NY, USA: ACM; 1999. p. 28–36.
- Kong A, Cheung K, Zhang D, Kamel M, You J. An analysis of biohashing and its variants. *Pattern Recognition* 2006;39(7):1359–68.
- Kotlarchyk A, Pandya A, Zhuang H. Simulation and experimental studies on fuzzy vault fingerprint cryptography. *International Journal of Knowledge-based and Intelligent Engineering Systems* 2008;12(5,6):305–17.
- Kulkarni J, Patil B, Holambe R. Orientation feature for fingerprint matching. *Pattern Recognition* 2006;39(8):1551–4.
- Li J, Yang X, Tian J, Shi P, Li P. Topological structure-based alignment for fingerprint fuzzy vault. In: *19th international conference on pattern recognition, 2008, ICPR 2008*. p. 1–4.
- Lumini A, Nanni L. An improved biohashing for human authentication. *Pattern Recognition* 2007;40(3):1057–65.
- MacWilliams F, Sloane N. *The theory of error-correcting codes*. Amsterdam: North-Holland; 2003.
- Maio D, Maltoni D, Cappelli R, Wayman J, Jain A. FVC2002: second fingerprint verification competition. In: *International conference on pattern recognition*, vol. 16. p. 811–4.
- Maltoni D, Jain A, Prabhakar S. *Handbook of fingerprint recognition*. Berlin: Springer; 2009.
- Nagar A, Nandakumar K, Jain A. Securing fingerprint template: fuzzy vault with minutiae descriptors. In: *19th international conference on pattern recognition, 2008, ICPR 2008*. p. 1–4.
- Nandakumar K, Jain A. Multibiometric template security using fuzzy vault. In: *2nd IEEE international conference on biometrics: theory, applications and systems, 2008, BTAS 2008*. p. 1–6.
- Nandakumar K, Jain AK, Pankanti S. Fingerprint-based fuzzy vault: implementation and performance. *IEEE Transactions on Information Forensics and Security* 2007;2(4):744–57.
- Nanni L, Lumini A. An experimental comparison of ensemble of classifiers for biometric data. *Neurocomputing* 2006a;69(13–15):1670–3.
- Nanni L, Lumini A. Human authentication featuring signatures and tokenised random numbers. *Neurocomputing* 2006b;69(7–9):858–61.
- Nanni L, Lumini A. Random bands: a novel ensemble for fingerprint matching. *Neurocomputing* 2006c;69(13–15):1702–5.
- Nanni L, Lumini A. A hybrid wavelet-based fingerprint matcher. *Pattern Recognition* 2007;40(11):3146–51.
- Nanni L, Lumini A. Local binary patterns for a hybrid fingerprint matcher. *Pattern Recognition* 2008a;41(11):3461–6.
- Nanni L, Lumini A. Random subspace for an improved biohashing for face authentication. *Pattern Recognition Letters* 2008b;29(3):295–300.
- Nanni L, Lumini A. Descriptors for image-based fingerprint matchers. *Expert Systems with Applications* 2009;36(10):12414–22.
- Nilsson K, Bigun J. Localization of corresponding points in fingerprints by complex filtering. *Pattern Recognition Letters* 2003;24(13):2135–44.
- Ram S, Bischof H, Birchbauer J. Modelling fingerprint ridge orientation using Legendre polynomials. *Pattern Recognition* 2010;43(1):342–57.
- Ross A, Jain A, Reisman J. A hybrid fingerprint matcher. *Pattern Recognition* 2003;36(7):1661–73.
- Sherlock B, Monro D. A model for interpreting fingerprint topology. *Pattern Recognition* 1993;26(7):1047–55.
- Shi Z, Govindaraju V. A chaincode based scheme for fingerprint feature extraction. *Pattern Recognition Letters* 2006;27:462–8.
- Stallings W. *Cryptography and network security: principles and practice*, 4th ed. Englewood Cliffs, NJ: Prentice-Hall; 2005.
- Tico M, Kuosmanen P. Fingerprint matching using an orientation-based minutia descriptor. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2003;25(8):1009–14.
- Uludag U, Jain A. Securing fingerprint template: fuzzy vault with helper data. In: *Proceedings of CVPR workshop on privacy research in vision*. IEEE Computer Society; 2006. p. 163.
- Wang Y, Hu J, Phillips D. A fingerprint orientation model based on 2d fourier expansion (fomfe) and its application to singular-point detection and fingerprint indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2007;29(4):573–85.
- Wegman M, Carter L. New hash functions and their use in authentication and set equality. *Journal of Computer and System Sciences* 1981;22(3):265–79.
- Yang S, Verbauwhe I. Automatic secure fingerprint verification system based on fuzzy vault scheme. In: *IEEE international conference on acoustics, speech, and signal processing, 2005. Proceedings (ICASSP'05)*, vol. 5.
- Zhou J, Gu J. A model-based method for the computation of fingerprints' orientation field. *IEEE Transactions on Image Processing* 2004;13(6):821–35.