Pattern Recognition Letters 45 (2014) 189–196

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Fingerprint shell: Secure representation of fingerprint template ☆

Chouaib Moujahdi [a,*], George Bebis [b], Sanaa Ghouzali [c], Mohammed Rziza [a]

[a] LRIT, Associated Unit to CNRST (URAC 29), Mohammed V-Agdal University, Rabat, Morocco
[b] Department of Computer Science and Engineering, University of Nevada, Reno, NV 89577, USA
[c] Information Technology Department, CCIS, King Saud University, Riyadh, Saudi Arabia

ARTICLE INFO

Article history:
Received 6 September 2013
Available online 13 April 2014

Keywords:
Fingerprint
Minutiae
Template protection
Revocability
Diversity
Security

ABSTRACT

Fingerprint is a popular biometric modality which is used extensively in several applications for person authentication, providing high uniqueness and acceptable performance. Most fingerprint systems use minutiae-based representations. However, several studies have proven that the original fingerprint impression can be reconstructed from minutia information, which makes the problem of ensuring the security of fingerprint data very critical. In this paper, we present a new approach for fingerprint template protection. Our objective is to build a non-invertible transformation that meets the requirements of revocability, diversity, security and performance. In this context, we exploit the information provided by the extracted minutiae to construct a new representation based on special spiral curves, which can be used for the recognition task instead of the traditional minutiae-based representation. The proposed approach has been evaluated using the original FVC protocol and compared with existing protection approaches which use the same protocol. Our experimental results illustrate the ability of the proposed representation to preserve the performance of protected systems. Moreover, we demonstrate that the security of our approach is sufficiently robust to the zero effort and brute force attacks.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Biometric systems, day after day, propagate more to human life instead traditional systems which use passwords and ID cards. They are widely used to identify/authenticate users reliably in many applications. However, biometric systems have given rise to new problems and challenges related to the security and the protection of personal data, issues of less concern in traditional systems. In practice, while biometrics ensure uniqueness, they do not provide secrecy. Each person has his/her own biometric traits but the person cannot keep them away from theft incidents to be used illegally. For example, a person might leave his/her fingerprints on everyday touched surfaces. Thus, many attacks can be launched against biometric systems, which reduce the credibility of these systems.

We can identify eight levels of attack in a biometric system [19]; however, since the principle of some attacks is repeated, Jain et al. [11] have grouped them into four categories. First, attacks on the user interface (i.e., sensor), mainly due to the presentation of falsified biometric data. Second, attacks on the interface between modules where an adversary can either destroy or interfere communication interfaces between modules. Third, attacks on the software module where the executable program of a module can be modified so that it always returns the desired values of an opponent. This is known as *Trojan-horse* attack.

Finally, attacks against the biometric templates stored in the database module which are considered among the most damaging attacks on a biometric system. For example, a biometric template can be replaced by an impostor's template to obtain unauthorized access to the system. In addition, a physic parody (spoof) can be created from a stolen template to obtain unauthorized access to the system. The irrevocability of biometric templates makes this kind of attack very dangerous; because, unlike a stolen credit card or password, it is not possible for a legitimate user to revoke his/her biometric templates and replace them with another set of identifiers. Therefore, promoting the use of biometric technologies in future applications requires increased security of biometric data.

Due to these security challenges, there are currently many research efforts underway to protect biometric systems against possible attacks. Several approaches have been proposed in the literature for biometric template protection. The main objective

☆ This paper has been recommended for acceptance by Ajay Kumar.

* Corresponding author. Address: 590 Lake Street, Reno, NV 89501, USA.
E-mail addresses: chouaib.moujahdi@fulbrightmail.org (C. Moujahdi), bebis@cse.unr.edu (G. Bebis), sghouzali@ksu.edu.sa (S. Ghouzali), rziza@fsr.ac.ma (M. Rziza).

http://dx.doi.org/10.1016/j.patrec.2014.04.001
0167-8655/© 2014 Elsevier B.V. All rights reserved.

of these proposed schemes is to make biometrics revocable. Revocability means that we can revoke a compromised template and replace it with another one, in the same way that a stolen password can be replaced with a new one. The main idea of these approaches, which aim to protect the stored biometric templates, is that instead of storing the templates themselves, a function is stored for each template which is used directly in the task of authentication. This work is primarily concerned with these solutions for template protection. An ideal approach of biometric template protection must meet four requirements [11]:

- *Revocability*: it should be possible to revoke a compromised template and replace it with a new one based on the same biometric data.
- *Diversity*: if a revoked template is replaced by a new model, it should not correspond with the former. This property ensures the privacy.
- *Security*: it must be difficult, computationally, to obtain the original template from the protected template. This requirement has another naming, for example it is called *non-reversibility* in [15] and *irreversibility* in [4]. It should be noted that the security requirement of a biometric template protection differs from the concept of the *overall security* of a biometric system which is the result of including all possible protection techniques in a biometric system (i.e., encryption of data in canals, timestamp, etc.) to avoid all possible threats in all levels of attack.
- *Performance*: The protection approach should not degrade the recognition performance of the system.

Jain et al. [11] have classified template protection approaches into three main categories: *feature transformation*, *biometric cryptosystem*, and *hybrid*. Each of these approaches has its own advantages and limitations [20]. Overall, they do not meet, contemporaneously, the four requirements of an ideal protection scheme. We are concerned in this work with feature transformation approaches. The basic idea of feature transformation approaches is to apply a transformation function F to the original biometric template T using a key K; the transformed template F(T,K) is then stored in the database. The function F is also used to transform the test template Q, and we can directly compare the transformed templates F(T,K) and F(Q,K) in the transformation domain to determine whether the user is accepted or not. Depending on the type of template representation, feature transformation schemes can be divided into two main classes: *Vector-based* approaches (e.g., [17] and *Interest-point-based* approaches (e.g., [18] (mostly designed for minutiae-based systems).

In this paper, we propose a new non-invertible transformation approach for *minutiae-based* templates of fingerprint-based systems (i.e., interest-point-based approach), that allows diversity, revocability, security and performance. In addition, unlike several fingerprint template protection schemes, the performance of our approach is less sensitive to translation/orientation transformations of fingerprint impressions.

The rest of the paper is organized as follows. In Section 2, we review fingerprint template protection approaches. We present the proposed approach in Section 3. Our experimental results and comparisons are presented in Section 4. Our conclusions and perspectives are provided in Section 5.

2. Literature review

Fingerprint recognition of minutiae-based systems involves the following three main steps. First, fingerprint image pre-processing (i.e., segmentation, orientation estimation, binarization, thinning, etc.). Second, feature extraction (ridge endings and bifurcations). Finally, fingerprint matching (i.e., alignment, matching score). This kind of systems requires storing minutiae information (coordinates and orientations) in the database. However, several works [21,6] have proven that the fingerprint impression can be reconstructed from minutia information. Thus, the design of new protection solutions and new secure fingerprint representations become increasingly important.

In practice, intra-subject variations make fingerprint recognition an extremely difficult task; this is because multiple acquisitions of the same finger are very unlikely to lead to an identical set of minutiae. The main factors responsible for intra-subject variations are due to the non-linear distortion (due to the skin elasticity), translation and rotation of fingerprint impressions. A finger may be placed and/or rotated on the sensor differently during several authentications which changes drastically the location/orientation information of minutiae points and requires applying an alignment between test and training templates before matching (e.g., according to [15], displacement of 2 mm corresponds to 40 pixels translation and $\pm 20^{\circ}$ of rotation can be noticed). Therefore, fingerprint recognition is very sensitive to orientation and translation of impressions (Fig. 1) which makes fingerprint template protection more complicated too.

Several *vector-based* approaches (e.g., [22,17] have been applied for fingerprint template protection. The main idea of these schemes is to build feature vectors using the global texture of fingerprint impressions. However, fingerprint images are mostly treated using an interest point representation (i.e., minutiae representation) which requires appropriate protection techniques. We can divide *interest-point-based* approaches into three categories.

First, techniques that convert the minutiae representation to a vector representation. For example in [8], the information provided by several extracted minutiae triplets are used to generate a binarized histogram. A user's key is used after that to randomize the histogram and obtain the protected binary vector to be stored in the database. This technique provides good performance. However, it is still vulnerable against some attacks like dictionary based attacks. In [23], several symmetric polynomial functions are used to construct the hashes of the extracted minutiae triplets during enrolment and verification stage; the matching takes place in the hash space. This technique provides good balance between security and performance. However, in practice, the hash values, of several impressions from the same finger, change drastically due to the presence of the intra-subject variations which can decrease considerably the performance. Boult et al. [3] propose a hybrid approach which combines feature transformation with encryption to generate a secure template called *Biotop biotoken*. This method provides a good balance between security and performance. Ahn et al. [2] propose an interesting alignment-free feature transformation approach. The purpose of this technique is to extract some special geometrical information from minutiae triplets to construct the secure template. This technique provides low accuracy compared to the state-of-the-art. Kumar et al. [12] propose an extension of [23] using a combination of symmetric hash functions on several extracted minutia k-plets. This technique provides high security in terms of resistance against brute force attacks. However, the performance is decreased.

Second, techniques that disorder the minutiae representation to generate a new secure set of minutiae [18,1]. Ratha et al. [18] have proposed an interesting solution which belongs to this category. The main idea is to apply geometric transformations to the minutiae representation. Three transformation types were tested: *cartesian*, *radial* and *functional*. This solution provides high security because it is difficult to recover the original minutiae representation from

the transformed template. However, the intra-subject variation increases in the secure representation which decreases the performance considerably.

Finally, techniques that generate a new secure representation from the minutiae representation. For example in [9], they have proposed a protected version of Minutia Cylinder-Code (MCC) [5] which is a new fingerprint template representation. MCC uses, for each minutia, a cylinder (i.e., local descriptor) to encode, spatial (location) and directional (orientation) information between the minutia and its neighborhood, and create the fingerprint template. A non-invertible transformation based on the *Kullback–Leibler* projection [10] followed by a binarization step is applied on the MCC. This scheme has demonstrated very good efficiency although it has to trade performance/accuracy for security/privacy. Our work is primarily concerned with this kind of solutions.

- Calculate the distance between each minutiae and *every* singular point (i.e., for every fingerprint template, the number of curves will be equal to the number of singular points). The distances between minutiae and singular point are invariant under translation or/and rotation transformations (Fig. 1).
- *Fingerprint shell/curve construction*: Sort distances in an ascending order. The sorted distances are used to construct several *contiguous right angle triangles* where the distances are the *hypotenuses* of these triangles (see Fig. 2). We keep only the spiral curves for matching.

It should be noted that for the first triangle, we choose randomly an initial distance d_0 (see Fig. 2(a) and Algorithm 1); for the other *cathetus*, the distance of the *leg* is calculated using the *Pythagorean theorem*. Moreover, d_0 is added to each extracted

Algorithm 1. Fingerprint curve construction

Input: Sorted distances $d_1\ d_2 \ldots\ d_n$ of a fingerprint impression
Parameters: User's key d_0
Output: A fingerprint curve *FC*
Let $DIS = [d_0,\ d_1 + d_0,\ d_2 + d_0, \ldots,\ d_n + d_0]$ and $\theta_1 = 0$
for $i = 1$ to $n + 1$ **do**
  if $i == 1$ **then**
    $Leg_i = \sqrt{DIS_{i+1}^2 - d_0^2}$ ▷ Leg distance of the first right angle triangle
    $x = [0\ DIS_i\ DIS_i\ 0]$ ▷ Points abscissae of the first triangle
    $y = [0\ 0\ Leg_i\ 0]$ ▷ Points ordinates of the first triangle
  endif
  if $i > 1$ **then**
    $Leg_i = \sqrt{DIS_{i+1}^2 - DIS_i^2}$ ▷ Leg distance of the next right angle triangle
    $\theta_i = -atan\left(\frac{Leg_{i-1}}{DIS_{i-1}}\right)$
    $\theta_i = \theta_i + \theta_{i-1}$ ▷ Angle between d_0 and the last constructed leg
    $x = [0\ DIS_i\ DIS_i\ 0]$
    $y = [0\ 0\ Leg_i\ 0]$
    for $j = 1$ to 4 **do**
      $\Delta = \begin{bmatrix} x_j & y_j \end{bmatrix} \times \begin{bmatrix} cos(\theta_i) & -sin(\theta_i) \\ sin(\theta_i) & cos(\theta_i) \end{bmatrix}$ ▷ Change of basis
      $x_j = \Delta_1$ and $y_j = \Delta_2$ ▷ Coordinates in the vector space frame of the first triangle
    end for
  end if
  $CurveX_i = x_3$ and $CurveY_i = y_3$
end for
$FC(1,:) = CurveX$ ▷ Points abscissae of the fingerprint curve
$FC(2,:) = CurveY$ ▷ Points ordinates of the fingerprint curve

3. Proposed approach

The main idea of fingerprint shell is to construct special *spiral curves* using information from the extracted minutiae. These curves will be stored in the database system to be used for recognition. This new representation of fingerprint images provides: revocability, diversity, security and performance. The main steps of fingerprint shell are the following:

During enrollment, for each fingerprint image we perform the following:

- Extract minutiae and *singular points*[1] (*Core* and *delta*).

[1] Singular points are special regions in a fingerprint impression where the ridges are of high curvature. These regions may be classified into two categories: *core* and *delta*. The number of singular points in a fingerprint template is usually between one and four points.

distance, before the triangles construction process, to be able to launch our algorithm even in the scenario where d_0 is greater than some extracted distances.

Each user will have his own d_0 which can be considered as the *user's key*. In practice, performance increases considerably using this strategy due to the reduction of the False Match Rate (see Section 4.2).

During authentication, for each test image, we use the same process applied on the training fingerprint images except for the choice of the singular point. For test templates, the closest singular point to the center of the test image is chosen to calculate the distances. However, any other singular point can be used in this process since in the enrollment phase we have considered all extracted singular points.

To match the test and reference curves, we apply the *Hausdorff Distance* which is widely used in computer vision and computer graphics. For two sets of points A and B, it can be defined as follow:

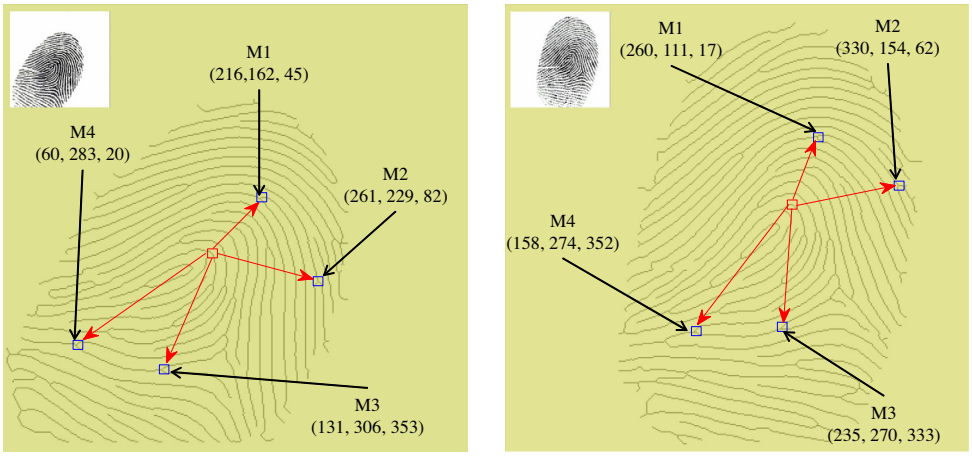
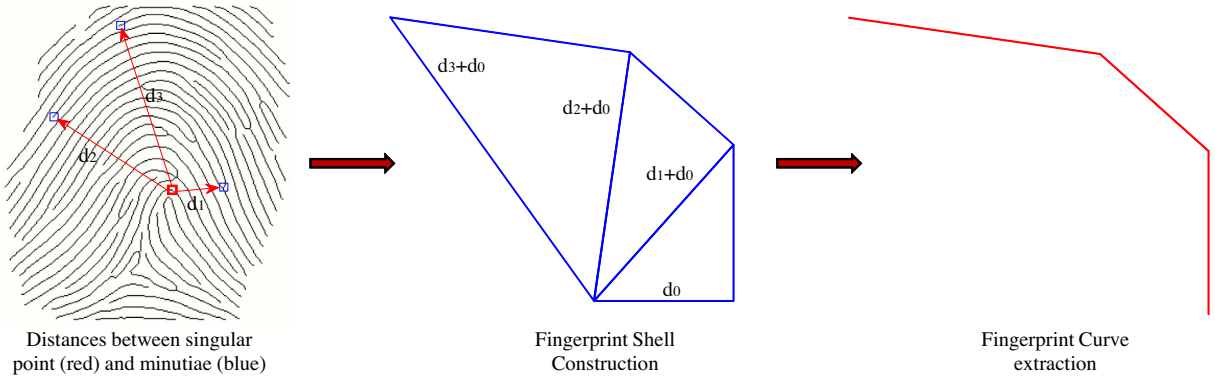
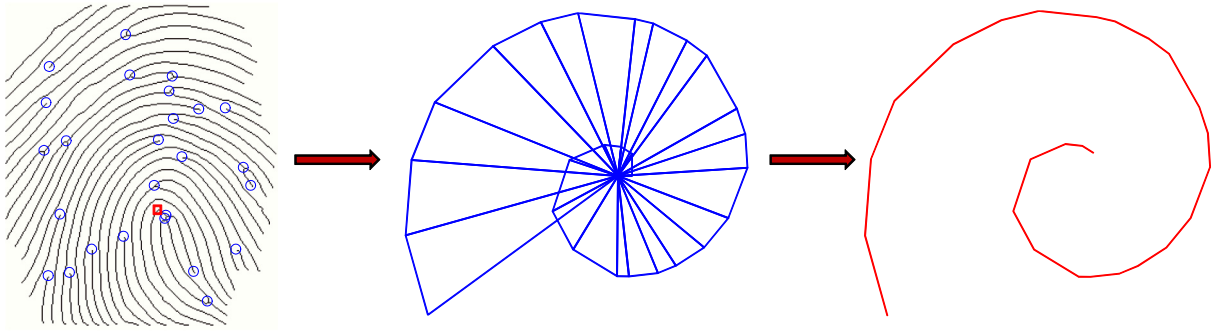


Fig. 1. Examples of translation/rotation of two impressions from the same finger. Location and orientation of minutiae (x, y, θ) are changed drastically. Distances between singular point (red square) and minutiae (blue squares) are relatively invariant. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(a) Simple example of Fingerprint Shell/Curve construction



(b) Fingerprint Shell/Curve of a fingerprint impression

Fig. 2. Illustration of Fingerprint shell construction.

$$HD(A,B) = max(h(A,B), h(B,A)) \quad (1)$$

where $h(A,B) = \max_{a \in A} \min_{b \in B} \|a - b\|$

– $\|.\|$ is a distance metric (e.g., Euclidean distance).

The proposed technique meets the requirements of revocability, diversity and security. In practice, we can protect a compromised template by changing the distance d_0 (revocability); the new constructed curve will not match with the compromised one which provides diversity (see Fig. 3).

For security (i.e., non-reversibility), it is very difficult to recover the minutiae information (which will be used as well to recover the fingerprint impression [21,6]) from a stolen fingerprint curve. Let us analyze the worst case scenario where the adversary has access to a fingerprint curve and he/she knows the process to recover the distances between the singular point and the extracted minutiae. In practice, the number of possibilities to put

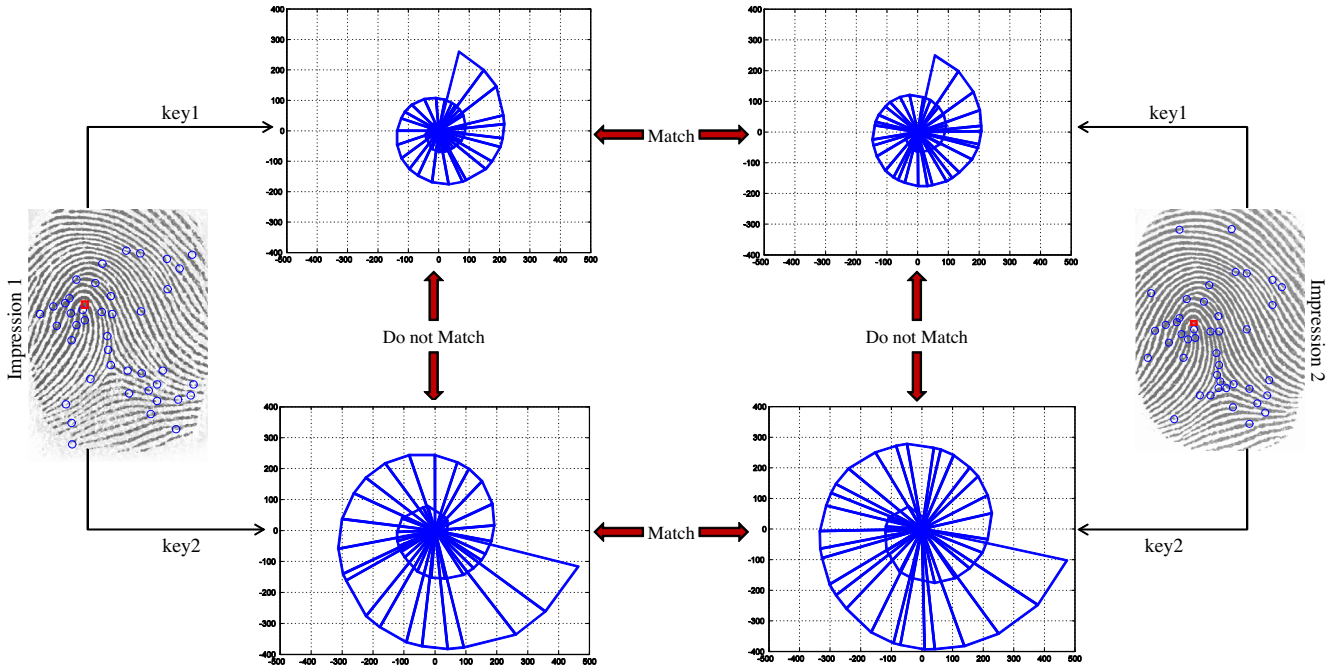


Fig. 3. Illustration of revocability/diversity; example of two impressions of the same identity using two different d_0.

each minutia around the singular point using the recovered distances is infinite. However, even if we assume that, for each distance, there are only 360 possibilities, the number of possible combinations is 360^n (n is the number of minutiae). Moreover, there is no way to recover the minutiae orientation, from a stolen fingerprint curve, which is necessary to reconstruct the fingerprint impression.

Based on the complexity of brute force attack, we can conclude that even in the worst case scenario, the security of our system is enough to be robust to this kind of attacks. The performance, diversity and security of the proposed approach will be analyzed more in the next section.

4. Experimental results

In this section, we evaluate the verification accuracy of the Fingerprint shell approach using the original Fingerprint Verification Competition protocol [14] and the FVC2002 DB1 and DB2 fingerprint databases.

4.1. Data sets and experimental procedure

FVC2002 DB1 and DB2 contain 800 fingerprint impressions, of various quality, from 100 distinct fingers (i.e., each person is represented by 8 impressions). The trial version of the commercial software VeriFinger SDK 6.0[2] has been used to extract minutiae and singular points.

For each database, the FVC protocol is used to report results of the Fingerprint shell on FVC2002. In this protocol, the first impression of each finger is compared against the first impression of the remaining fingers to obtain the impostor score distribution (i.e., 4950 attempts in the case of the FVC2002 databases if all first impressions are enrolled successfully). To obtain the genuine score distribution, each impression is compared against the remaining impressions of the same finger (i.e., 2800 attempts in the case of the FVC2002 databases if all impressions are enrolled successfully).

[2] http://www.neurotechnology.com

It should be noted that the calculated scores must be in the range [0,1] and the symmetric comparisons are not launched to avoid repetition of scores (i.e., if T_1 is matched with T_2, T_2 against T_1 is not calculated).

The genuine/impostor score distribution can be graphically illustrated to show how an algorithm can separate imposter from genuine. We will use two other factors to measure the separability of scores: first, the Kolmogorov–Smirnov test. The closer this test is to 1, the more the scores are separated, which means that diversity is high. Second, the separability measurement proposed by Lee et al. [13]:

$$Separability = \frac{|\mu_G - \mu_I|}{\sqrt{(\sigma_G^2 + \sigma_I^2)/2}} \tag{2}$$

- μ_G and μ_I are the means of genuine and impostor distributions.
- σ_G^2 and σ_I^2 are the variances of genuine and impostor distributions.

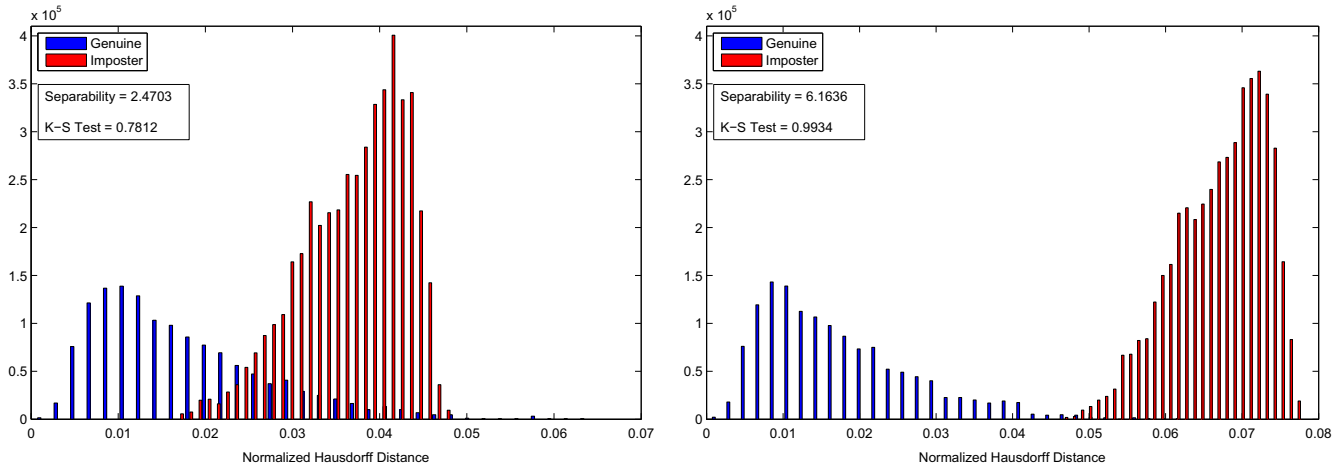
Following the FVC protocol, genuine matching scores (*gms*) and impostor matching scores (*ims*) are used to calculate the False Match Rate (FMR) and the *corrected* False Non Match Rate (FNMR). For a threshold t ranging from 0 to 1 [14]:

$$FMR(t) = \frac{cardinality\{ims | ims \geqslant t\}}{number\ of\ impostor\ recognition\ attempts} \tag{3}$$

$$FNMR(t) = \frac{cardinality\{gms | gms < t\} + REJ}{number\ of\ genuine\ recognition\ attempts} \tag{4}$$

- Where *REJ* is the number of rejections. If an image cannot be enrolled successfully (e.g., no extracted singular point in the case of our approach), the matching will be 0 for every possible recognition attempt using any rejected template.

Additional performance indicators are used in our evaluation [7]: first, FMR_{1000} which equal the FNMR value when FMR = 0.001%. Second, $Zero_{FMR}$ which equal to the lowest FNMR



(a) Without a user's key (b) With a user's key

Fig. 4. Histogram of two systems based on fingerprint shell using FVC2002 DB1.

value for FMR = 0%. These values are used to evaluate the verification accuracy of systems which operate far from the Equal Error Rates (EER) point (i.e., these systems aim high security and they use a threshold which reduces FMR even if that yields a high FNMR).

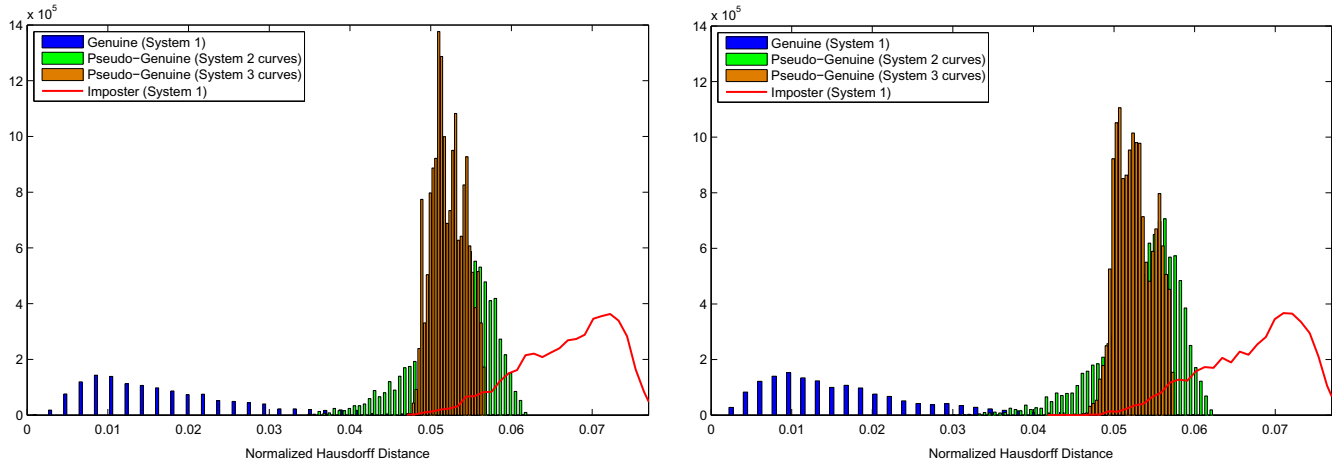
Fingerprint curves have been constructed as described in Section 3. However, to use correctly the FVC verification protocol and allow a fair comparison with the state-of-the-art, we only extracted the closest singular point to the image center to calculate distances. It should be noted that three images from DB1 and one image from DB2 do not contain singular points; which means that the number of possible recognition attempt using these rejected images, among the 2800 attempts of genuine scores, is 21 and 7 for DB1 and DB2 respectively. Therefore, in our evaluation, $REJ_{DB1} = 0.75\%$ and $REJ_{DB2} = 0.25\%$.

4.2. Results and discussion

First, we evaluated the importance of using a specific d_0 for each user. Separability, Kolmogorov–Smirnov test and genuine-impostor distribution of two systems, using FVC2002 DB1 database, are reported to show how the use of keys influences discriminability. The first system enrolls people without a user's key (i.e., the same d_0 is used for all fingers). In the second system, the user's key is required (the keys are randomly chosen in the range $(0, 1.5555]$). Fig. 4 illustrates the results.

Fig. 4(a) shows the genuine-impostor distribution of the first system (i.e., without keys). We can notice some overlap between the two distributions which explains the medium values of K–S test (0.7812) and Separability (2.4703). These values are greatly increased (6.1636 and 0.9934 for Separability and K–S test respectively) using the user's keys in the second system (Fig. 4(b)) which means that discriminability is enhanced as well. Therefore, we can conclude that the use of a specific d_0 for each user increases discrimination which reduces the False Match Rate and increases accuracy.

The purpose of our second evaluation is the diversity of fingerprint shell approach (i.e., resistance against cross-matching). It must be ensured that a fingerprint curve of a finger does not allow cross-matches among fingerprint curves of the same finger in other systems. For each database, we have considered three systems which enroll the same fingers in their databases. Systems 1, 2 and 3 use keys which are randomly chosen in the ranges $(0, 1.5555]$, $[100, 500]$ and $[1000, 2000]$ respectively. For each finger in system 1, the same finger in the systems 2 and 3 is used to



(a) FVC2002 DB1 (b) FVC2002 DB2

Fig. 5. Histogram of system 1 and pseudo-genuine distributions using systems 2 and 3.

Table 1
Verification accuracy based on FVC protocol (percentage values).

	FVC2002 DB1			FVC2002 DB2		
	EER	FMR_{1000}	$Zero_{FMR}$	EER	FMR_{1000}	$Zero_{FMR}$
[3]	2.1	–	–	1.2	–	–
[23]	3	–	–	–	–	–
[2]	7.18	–	–	3.61	–	–
[12]	–	–	–	4.98	–	–
[9]	1.88	3.14	5.07	0.99	1.43	2.54
Fingerprint shell	2.03	4.18	6.36	1.01	1.39	2.21

obtain the *pseudo-genuine* distribution as follow: each curve from system 1 is compared against all curves of the same finger from systems 2 and system 3 respectively (i.e., 6400 attempts for each system if all impressions are enrolled successfully; 6355 attempts for DB1 and 6385 attempts for DB2 in our experiments). The results are illustrated in Fig. 5.

Fig. 5 shows the genuine-impostor distribution of system 1 and the pseudo-genuine distributions in the same system using fingerprint curves from system 2 and system 3. It can be observed that both pseudo-genuine distributions are well separate from the genuine distribution of system 1. Moreover, they are closest to the impostor distribution which means that system 1 considers, most of time, fingerprint curves of systems 2 and 3 as impostors. Thus, we can conclude that both revocability and diversity are achieved by the proposed approach. We can also notice that the pseudo distribution of system 3 is more separated from the genuine distribution of system 1 than that of system 2; which means that the diversity, of two systems based on fingerprint shell approach, increases if the keys ranges are well separated.

Table 1 provides a comparison of verification accuracy of the fingerprint shell approach with existing protection approaches which use the same FVC protocol. It should be noted that the fingerprint shell and [3], unlike the other algorithms cited in Table 1, are *two-factor* techniques which combine the information provided by a fingerprint impression with a secret key (for fingerprint shell, the keys are randomly chosen in the range $(0,100]$).

We can notice that the proposed approach shows good performance in comparison with the state-of-the-art, which can be explained by the increase of discrimination/performance (which is one of the advantages of two-factor approaches [15] due to the use of a specific key for each user and also by the fact that all techniques of Table 1 (except [3] are single-factor approaches. In addition, the use of information invariant to translation/rotation of impression (i.e., distances between singular points and minutiae) helps to preserve performance.

Table 2
Verification accuracy of Fingerprint shell in the zero effort attack scenario.

	FVC2002 DB1			FVC2002 DB2		
	EER	FMR_{1000}	$Zero_{FMR}$	EER	FMR_{1000}	$Zero_{FMR}$
Fingerprint shell	4.28	27.14	94.00	1.45	36.46	99.04

To analyze the performance of Fingerprint shell in the zero effort attack scenario (with stolen key) where the opponent knows d_0 and tries to circumvent the system using his/her own fingerprint features, we modified the FVC evaluation protocol. To obtain the impostor score distribution, the first fingerprint curve of each finger is compared against the first curve of the remaining fingers which is constructed using the same d_0 of the reference curve (i.e., 9900 attacks if all first impressions are enrolled successfully). For the genuine score distribution, each impression is compared against the remaining impressions of the same finger (i.e., 2800 attempts if all impressions are enrolled successfully). It should be noted that the keys are randomly chosen here in the range $(0,1000]$. The ROC curves of fingerprint shell using the described scenario are illustrated in Fig. 6.

We can notice that the proposed approach preserves, in general, the performance of the protected systems which means that fingerprint shell resists against the zero effort attacks even in the case where the adversary knows the user's key. However, we can observe from Table 2, which summarizes the EER, FMR_{1000} and $Zero_{FMR}$ computed from Fig. 6, that the performance is preserved just near the EER points (4.28% and 1.45% for DB1 and DB2 respectively). Fingerprint shell loses accuracy for thresholds far from the EER points (see FMR_{1000} and $Zero_{FMR}$ in Table 2), which can be explained by the inability of the used classifier to return a score which exceeds the threshold decision in high level security working points.

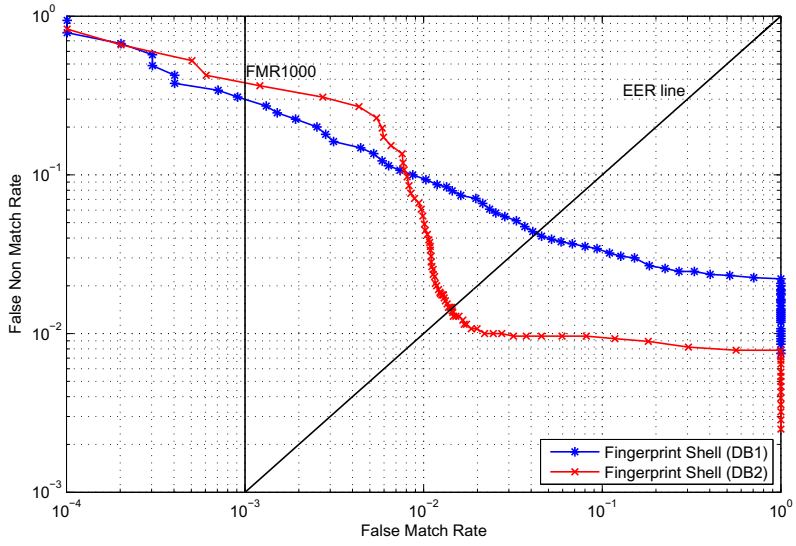


Fig. 6. ROC curves in the zero effort attack scenario using FVC2002 DB1 and DB2.

Therefore, we can conclude that, in high security applications based on fingerprint shell, the systems should be operated only near the EER point to minimize the success of zero effort attacks.

5. Conclusion

In this paper, we proposed a new approach for fingerprint template protection. The information provided by the minutiae is used to construct a new representation based on special spiral curves which are used for the recognition task instead of the traditional minutiae-based representation. Our approach meets revocability, diversity and security, which are required in an ideal method for template protection. In addition, our experimental results indicate that the proposed approach preserves recognition performance. The performance of fingerprint shell is related to two factors: first, the accuracy of minutiae extraction and second, the presence and accuracy of singular points detection. However, unlike several fingerprint template protection schemes, the performance of fingerprint shell is less sensitive to translation/rotation of fingerprint impressions.

In practice, the proposed technique is far from being efficient in all scenarios of impression acquisition: for example, if the majority of minutiae are missed or several spurious minutiae are added or simply the impression is cropped, the constructed curves will change drastically. In our future work, we will address these weak points. In addition, we plan to test the proposed approach using larger databases which are characterized by the presence of important intra-subject variations. Also we plan to improve fingerprint shell using more features provided by fingerprint impressions (e.g., texture measures) to construct the curves, and to test several versions of the Hausdorff distance family and new methodologies of this distance (e.g., [16]. Moreover, our future plans include the design of new protection schemes for multimodal systems.

Acknowledgment

The first author is a fulbright visiting scholar, to University of Nevada-Reno, of the Moroccan-American Commission for Educational and Cultural Exchange. Dr Sanaa Ghouzali is grateful for the support of the Research Center of College of Computer & Information Sciences and the Deanship of Scientific Research, King Saud University, under grant RC120901.

References

[1] T. Ahmad, J. Hu, S. Wang, Pair-polar coordinate-based cancelable fingerprint templates, Pattern Recognit. 44 (2011) 2555–2564.
[2] D. Ahn, S. Kong, Y.S. Chung, K.Y. Moon, Matching with secure fingerprint templates using non-invertible transform, in: Congress on Image and Signal Processing, CISP '08, 2008, pp. 29–33.
[3] T. Boult, W. Scheirer, R. Woodworth, Revocable fingerprint biotokens: accuracy and security analysis, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR '07, 2007, pp. 1–8.
[4] J. Breebaart, B. Yang, I.B. Dulman, C. Busch, Biometric template protection: the need for open standards, Privacy Data Secur. J. 5 (2009) 299–304.
[5] R. Cappelli, M. Ferrara, D. Maltoni, Minutia cylinder-code: a new representation and matching technique for fingerprint recognition, IEEE Trans. Pattern Anal. Mach. Intel. 32 (2010) 2128–2141.
[6] R. Cappelli, D. Maio, A. Lumini, D. Maltoni, Fingerprint image reconstruction from standard templates, IEEE Trans. Pattern Anal. Mach. Intel. 29 (2007) 1489–1503.
[7] R. Cappelli, D. Maio, D. Maltoni, J. Wayman, A. Jain, Performance evaluation of fingerprint verification systems, IEEE Trans. Pattern Anal. Mach. Intel. 28 (2006) 3–18.
[8] F. Farooq, R. Bolle, T.Y. Jea, N. Ratha, Anonymous and revocable fingerprint recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR '07, 2007, pp. 1–7.
[9] M. Ferrara, D. Maltoni, R. Cappelli, Noninvertible minutia cylinder-code representation, IEEE Trans. Inf. Forensics Secur. 7 (2012) 1727–1737.
[10] C. Goutis, C.P. Robert, Model choice in generalised linear models: a bayesian approach via Kullback–Leibler projections, Biometrika 85 (1998) 29–37.
[11] A.k. Jain, K. Nandakumar, A. Nagar, Biometric template security, EURASIP J. Adv. Signal Process. (2008) 1–17.
[12] G. Kumar, S. Tulyakov, V. Govindaraju, Combination of symmetric hash functions for secure fingerprint matching, in: 20th International Conference on Pattern Recognition (ICPR), 2010, pp. 890–893.
[13] C. Lee, J.Y. Choi, K.A. Toh, S. Lee, Alignment-free cancelable fingerprint templates based on local minutiae information, IEEE Trans. Syst. Man Cybern. Part B Cybern. 37 (2007) 980–992.
[14] D. Maio, D. Maltoni, R. Cappelli, J. Wayman, A. Jain, Fvc 2000: fingerprint verification competition, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2002) 402–412.
[15] D. Maltoni, D. Maio, A.K. Jain, S. Prabhakar, Handbook of Fingerprint Recognition, second ed., Springer Publishing Company Incorporated, 2009.
[16] R. Moreno, S. Koppal, E. de Muinck, Robust estimation of distance between sets of points, Pattern Recognit. Lett. 34 (2013) 2192–2198.
[17] C. Moujahdi, S. Ghouzali, M. Mikram, M. Rziza, G. Bebis, Spiral cube for biometric template protection, in: Image and Signal Processing, Lecture Notes in Computer Science, vol. 7340, Springer, Berlin Heidelberg, 2012, pp. 235–244.
[18] N. Ratha, S. Chikkerur, J. Connell, R. Bolle, Generating cancelable fingerprint templates, IEEE Trans. Pattern Anal. Mach. Intell. 29 (2007) 561–572.
[19] N. Ratha, J. Connell, R. Bolle, An analysis of minutiae matching strength, in: Audio- and Video-Based Biometric Person Authentication, Lecture Notes in Computer Science, vol. 2091, Springer, Berlin Heidelberg, 2001, pp. 223–228.
[20] C. Rathgeb, A. Uhl, A survey on biometric cryptosystems and cancelable biometrics, EURASIP J. Inf. Secur. 2011 (2011) 1–25.
[21] A.A. Ross, J. Shah, A.K. Jain, Toward reconstructing fingerprints from minutiae points, Proc. SPIE 5779 (2005) 68–80.
[22] A.B.J. Teoh, K.A. Toh, W.K. Yip, 2^n discretisation of biophasor in cancellable biometrics, in: Proceedings of the 2007 International Conference on Advances in Biometrics, Springer-Verlag, Berlin, Heidelberg, 2007, pp. 435–444.
[23] S. Tulyakov, F. Farooq, P. Mansukhani, V. Govindaraju, Symmetric hash functions for secure fingerprint biometric systems, Pattern Recognit. Lett. 28 (2007) 2427–2436.