Operational Bio-Hash to Preserve Privacy of Fingerprint Minutiae Templates

Rima Belguechi, Estelle Cherrier, Christophe Rosenberger, Samy Ait-Aoudia

Operational Bio-Hash to Preserve Privacy of Fingerprint Minutiae Templates

Rima Belguechi*, Estelle Cherrier, Christophe Rosenberger, Samy Ait-Aoudia

r_belguechi@esi.dz,estelle.cherrier@ensicaen.fr,christophe.rosenberger@ensicaen.fr, s_ait_aoudia@esi.dz

Abstract

The storage of fingerprints is an important issue as this biometric modality is more and more deployed for real applications. Considering minutiae templates as sensitive information, a key question concerns the secure and privacy management of this digital identity. Indeed, if an attacker obtains the minutiae template of an user, he/she will be able to generate a fingerprint having the same characteristics. Instead of directly storing the minutiae templates, we propose in this paper a new adaptation of BioHashing to generate a cancelable template in the context of un-ordered set of noisy minutiae features. To the authors knowledge, little interest has been paid in the literature to this question until now, since this is an acute problem. Using the FVC2002 benchmark database, we show the effectiveness of the proposed approach in term of privacy preservation. We show how the proposed method copes with irreversibility and diversity properties and therefore can be efficient in a realistic context.

Index Terms

Privacy, secure biometric systems, template protection

– – – – – – – – – – ☞ – – – – – – – – – –

I. Motivations and scope

In reference to information security, biometrics is applied to determine or verify the identity of an user. However, an industrial deployment of biometrics remains challenging as far as the security of the stored data is concerned. In a strong, world-wide consensus [1], it is admitted that the biometric identifier is personal information which raises serious privacy issues in case of abuse or identity theft. Furthermore, unlike passwords or tokens, if compromised, the biometric template

cannot be cancelled or revoked. Therefore biometric data is particularly sensitive. However, this claim has to be mitigated, since not all biometric deployments bear the same privacy risks. Fingerprint is considered highly risked since it is a tracing modality. It means that fingerprint data may be captured without knowledge nor consent, and hence used for undisclosed purposes.

Since the pioneering work of Ratha [26], one possible solution relies on a chosen transform enabling fingerprint template protection and thus, privacy protection.

Two approaches have been developed in the literature: biometric cryptosystems (see [32] for example and the references therein) and feature transformation based approaches [1]. In this paper, only the latter approach is considered.

In this article, a particular focus is made on the tokenised pseudo-random multispace transform called BioHashing [12]. It consists of a transformation of the biometric template into a compact vector code named biocode. The transform is based on an iterative inner product applied between a multispace matrix and the feature template. This matrix is generated from a random key K considered as a seed for a tokenized random number. This method is attractive for its different properties. For example, we can mention that, in zero knowledge scenario, BioHashing has been proven to be irreversible [10]. The irreversibility has a direct impact on preserving privacy. According to ISO/IEC 24745 '-Information technology Security techniques- Biometric information protection ', irreversibility prevents the use of biometric data from any other purpose than the originally intended one. It also guarantees a low or a null leakage of biometric information in the protected template. If the reference template is compromised, a direct replacement by a new one is possible by just issuing a new pseudo-random token. This renewability property aims to prevent tracking individual activities over different applications. This is fundamental for privacy preservation. Hence, BioHashing reveals itself as a promising solution to respect many privacy requirements. However, at the same time, it brings some research challenges to overcome. Certain properties only hold in the ideal case where no intruders is attacking the system. That is why some questions arise in a

more realistic context. When an impostor steals the key K, the accuracy is worse than when using the biometric alone. For example, in [39], Lumini and Nanni demonstrate that the performance in term of false acceptance rate (FAR) moves from 7.3% when no hashing is performed to 10.9% when BioHashing is operated under the hypothesis of stolen token. Talking of which, a relevant question is how to prevent the deterioration of the recognition accuracy when the genuine token is stolen and used by an impostor. A straightforward solution is to use a better feature extractor in hamming space (BioHashing requires a template as a fixed length feature vector) which will directly impact on the recognition performance. In [41], authors use the invariant LBP (Local Binary Pattern) texture operator. In their scheme, the fingerprint image is equally divided into sub-regions from which the LBP features are extracted. However, in their approach, authors make alignment using minutiae matcher which is not tolerable for protection purpose approaches. In [40], the author use a spectral fingerprint feature representation based on fourier transform. In stolen token scenario, the false acceptance rate is 7.31% on turkish government database. As we remark, the problem is that these descriptors are not sufficient. Considering fingerprint, it is well established that minutiae map is the moste accurate fingerprint representation. A better solution would be to adapt BioHashing to minutiae representation even if this is not obvious. Since the revelation of BioHashing in 2003, only few articles got interest on applying BioHashing to minutiae representation [14][34][35][36]. It is just in 2010 that works using minutiae in the BioHashing appear.

We show in this article how to adapt BioHashing to minutiae by using local matching. It consists of comparing two fingerprints according to the local descriptor of each minutia. Each local descriptor will be protected by BioHashing process. In the recent work of Nanni [36], the orientation descriptor of Tico [24] and a greedy matching algoritm were chosen. In parallel, Yang *et al.* [14] represents a minutia by a nearest neighbor based structures rather than local descriptor and protect this descriptor by BioHashing. In our work, we propose to use a texture descriptor based on the fingercode [2] where the matching is based on a neighborhood first search strategy, an enhanced

version of the greedy matching. Compared to Yang *et al.* [14], by using a local descriptor, we bring a greater code size which directly impacts on the security of the system. In the next, we show how our proposal fulfills diversity and randomness criteria. Compared to other works, we study these properties with a special care of the value of the decision threshold. This threshold is never considered by other works when it must be fixed in practice. The outline of the paper is as follows: Section 2 expounds fingerprint template transformation approaches. In Section 3 we present the proposed BioHashing model for minutiae. Sections 4 and 5 are devoted to evaluate the proposed system.

II. Related works on fingerprint template transformation approaches

Over the last decade, many attempts have been made to address the problem of protecting fingerprint templates. Jain *et al.* [1] classified the existing algorithms into biometric cryptosystems and transformation-based approaches.

In the first category, instead of directly encrypt the template; error correcting codes were designed as an alternative to deal with the variability of the biometric data. The main critical propositions of this category are : fuzzy commitment [4], fuzzy vault [3][8], secure sketch [32], biometric hardening password [21], shielding function [22], syndrome-based approach [33]. Multiple works have questioned the privacy-enhancing properties of these schemes. This is not our subject, but interested reader can refer to analyzes made in [37][38].

Approaches belonging to the second category make comparison directly in the transformation domain. Suppose the biometric X is transformed using a function F into the encoded data T during enrolment. For verification, the query biometric Y is encoded into the secret T' and the authentication will succeed if T is close to T' using a certain similarity distance.

A number of fingerprint transforms are belonging to this category. Cancelable biometrics proposed by Ratha et al. [25] applies different geometric transforms to minutiae points. However, in the recent work of Nagar *et al.* [7], linkalibility and irreversibility of these transforms have been really

disputed. Kumar *et al.* [19] propose a symmetric hash functions applied on triplets of neighboring minutia while cancelabillity is not considered. Farooq *et al.* [18] protect minutiae in the form of a histogram representation. Each minutiae triplet is represented by a 7 values invariant representation and binned in the form of an Histogram. Performance reported by authors is good but the system is complex to implement.Boult *et al.* [30] propose a transformation called biotope that induces a robust distance measurement. The biometric feature data X is first transformed (applied per feature) via scaling and rotation into $\grave{X} = (X - t) \times s$ while t and s are randomly generated parameters. The revocability is assured by the use of different values for t and s. With the biotope, authors can achieve a better accuracy than the baseline system. However, the robust revocable transform was calculated per user. The fair evaluation would be to transform all probes with the same transform and report the accuracies (stolen key scenario).

In parallel, techniques inspired from password salting are typically dual factor authentication where a user-specific key is introduced during the transformation. BioHashing [12] is the most representative method of this family. It is based on a linear random projection. Note that, featuring biometric data with user specific randomness like a key or a password seems to be the easiest way to achieve revocability via direct replacement with a new set of randomness.

The transformation function in BioHashing combines a user specific key K with the biometric feature expressed as a fixed-length vector $x=(x_1,..., x_n)/x\epsilon\mathbb{R}^n$. For more protection, the key K is considered as a seed for a tokenized random number (TRN). The BioHashing process is decoupled into two steps:

1) Random projection*:* It has been shown in [29] that random mapping can preserve the distances in the sense that the inner product between the mapped vectors closely follows the inner product of the initial vectors. The reference [29] proves that the closer to an orthonormal matrix the involved random matrix R is, the better the statistical characteristic of the feature topology are preserved. As a consequence, the tokenized random number is used as a seed to generate m random

vectors $r_j \epsilon \mathbb{R}^n / j = 1, \dots, m$ and $(m \leq n)$. After orthonormalization by the Gram-Schmidt method, these vectors are gathered as the column of a matrix $O = (O_{ij})_{i,j\epsilon[1,n]x[1,m]}$. The resulting vector is denoted $W = (W_1, \dots, W_m)\ ) / W =< O|x > \epsilon\ \mathbb{R}^m$.

2) Quantization: This step is devoted to the transformation in a binary-valued vector of the previous real-valued vector using a simple thresholding. The goal is to guarantee the irreversibility of the process. It requires the specification of a threshold τ_b to compute the final biocode b=(b_1,...,b_m) while $b_i = \{0\ if\ w_i \leq \tau_b, 1\ \text{else}\} / i = 1, \dots, m$. Thus, by combining the high confidence of the key to the biometric data, the inter-class variation increases while the intra-class distance is preserved. Hence, zero EER can be achieved even if the feature extractor is low. However, if the key is revealed to an impostor, performances become worse than when using the biometric alone [5].

The sustained works on BioHashing attempt to improve performance in when the user-specific TRN is stolen. This case is denoted 'stolen token scenario'.

In [11], authors use error correcting codes to resolve the stolen-token problem for PCA face features. The performance deterioration can be due to the information lost in binarization. In [9], the concept of multi-stage quantization is introduced. The idea is to represent each $b_i, \text{i}, \dots, m$ of the biocode by N_i bits. The choice of N_i depends on the user-standard deviation of $w_i =< x|r_i >$ where r_i are the column vectors of the random matrix. In [6], the improvement is based on fusion strategies. The authors exploit the observation that the biocode length impacts the performance in the basic approach. However, this length is bounded by the feature vector dimension and cannot be increased at will. So, they propose solutions to assign g biocodes per user (g>1) rather than one.

The use of BioHashing to points set like minutiae features is surprisingly limited even if we are certain that these features will give better results. As it is known, ridge features or other representation suffers from a lower performance compared to minutiae. However, the problem is that it is not obvious how to apply it to the minutiae set. In another hand, Using minutiae as template, we have to cope with some difficulties which are: i) Miss-alignment between two minutiae set, ii) Set

of minutiae points is not ordered and has variable size, iii) Missing or spurious minutiae.

Yang *et al.* [14] have presented a new method for self-alignment and protection of fingerprint templates. Each template is decomposed in a set of N minutiae vicinities. To each of these vicinities the BioHashing is applied. At the same time, Nanni *et al.* [36] protect by BioHashing the orientation descriptor of each minutia. In the next, we present our proposal in which a minutia is represented by its texture descriptor. The neighboring information is used to consolidate the local matching process. We point the relevance of our approach in term of irreversibility and diversity compared to existent methods.

III. Applying BioHashing to minutiae template

We propose in this work, two descriptors: texture-based descriptor, which captures the ridge flow patterns around each minutia, and minutiae-based descriptor, which reflects the relationships between each minutia and its neighborhood. We propose to protect the texture based descriptor by BioHashing while keeping the minutiae descriptor in clear.

Based on the proposed descriptors, a graph matching algorithm similar to [28] is used to establish the correspondences between minutiae.

We use the *libcubs library* provided by the Unified Biometrics and Sensors center (www.cubs.buffalo.edu) for minutiae extraction.

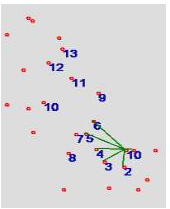
Our approach is as following:

Biometric descriptor creation: We propose to create minutia-centered local structures by using the texture descriptor presented in [2] around each minutia point *m*. We call this descriptor, a M*inuCode*. The M*inuCode* extraction requires enhancing the input image [28], and using a region of interest (ROI) determined by a circular tessellation surrounding this central point. This tessellation consists of *B* concentric bands of *b* pixels width. Each band is divided into 16 sectors of the same angle. The feature vector is calculated after application of a bank of Gabor filters using 8

orientations. The result is a vector of $m = B \times 16 \times 8$ elements. Fig.1.a shows the ROI.

Further, we use small sets of K-neighboring minutiae (K-Plet) as minutiae descriptor to represent the local structure between minutiae. We consider the K-Plet (Fig.1.b) as a reinforcement step in order to verify if the matching of texture descriptor is consistent at the global level. Hence, a K-Plet is formed by a central minutia *m* and its K spatially closest minutiae.

a. ROI around a minutia point



b. 6-plet of the central minutia labeled "1"

Fig.1. Biometric descriptor

The drawback in such neighbor-based structure is the possibility of exchanging nearest neighbors due to missing or spurious minutiae. To avoid this problem, two K-Plets are matched using a string alignement technique. We use the dynamic programming algorithm as proposed in [28]

Minutiae selection: given the fingerprint image T, we obtain the template minutiae set *M* such as $M = \{m_i\}_{i=1}^{\mu}$ with μ the number of minutiae in the image T. From the set M, a minutia m_i is selected only if its surrounding ROI is validated according to this principle: it is in the boundary of the image and each sector S represents an alternation of ridges and valleys. We express this alternation by the energy *E* of Fourier spectrum so, if $E > T_r$, then the sector is accepted else it is rejected, where T_r is a global Otsu threshold. It is clear that the final selected minutiae set S will be smaller than the initial one. In fact, usually there are 30~60 minutiae in one fingerprint. In our algorithm, we only select the minutia with valid ROI (implicitly the central ones), given a template of 10~20 minutiae, which is considered as sufficient when dealing with local structures [31].Therefore, we can adjust the ROI surface according to the minutiae count which in turn depends on the sensor attributes.

Rotation and translation support: while the translation factor is implicitly solved by using a reference point, the rotation problem is more critical. We adopt a solution based on the reference orientation correction. For each fingerprint image, we compute the reference orientation θ_{ref} using the method of Liu *et al.* [23] with some modifications. The orientation estimation is operated on the binarized image rather than the raw image. To avoid artifacts, we also consider θ_{ref}in the range of fundamental orientations as following:

$$\theta_{ref} \epsilon \left\{(k-1).\frac{\pi}{period}, k = 1,2, \dots, period\right\} \quad (1)$$

$period$ expresses the relevant accuracy for orientation angle. One can choose values of 16 or 32 to manage shifts of $\mp 11.25°$ or $\mp 5.625°$ resp. At matching step, the correction is done as following:

```
diff=min(Abs(ref1-ref2), 360- (Abs(ref1-ref2)))
if(ref1 < ref2)  diff=-diff;
imrotate(input_image, diff) (counterclockwise rotation)with ref1, ref2 the reference orientations
of the enrolled and input images resp.
```

Fig. 2. Rotation correction

protection: each minutia $s_i \in S, i = 1, \dots, \nu$ with ν the number of selected minutiae in the set *M* is represented by its MinuCode. After, a ν K-plets are created containing K MinuCodes for each one. To enhance privacy, we decide to protect each MinuCode by the BioHashing process explained in Fig. 3.

For each minutia $s_i \in S$
1. Let MinuCode the local descriptor of length n.
2. Normalize the MinuCode vector in the range [-1, 1] (the same range as the Gram-Schmidt result).
2. Let biocode a vector of length m.
3. Let K be the seed attributed to the user U.
4. Generate from K a uniform random matrix R_{nxm}.
5. Control that vectors in R are linearly independents else go to 4.
6. Apply the process of Gram-Schmidt to transform R to an orthonormal set. In this case, we must have: $m \leq n$.
7. Make the projection of MinuCode on this matrix:

$$[x_1, x_2, \dots x_n].\begin{bmatrix} or_{1,1} & \cdots & or_{m,1} \\ \vdots & \ddots & \vdots \\ or_{n,1} & \dots & or_{n,m} \end{bmatrix} = [w_1, w_2, \dots w_m]$$

8. Binarization of W by thresholding to obtain the biocode vector $[b_1, b_2, \dots b_m]$ such as:

$$b_i = \begin{cases} 0 & if \;\; w_i < \tau_b \\ 1 & else \end{cases}$$

τ_b is a binarization threshold. Its choice will be discussed just below.
9. Delete the MinuCode and store the biocode.

Fig. 3. BioHashing process pseudocode

The similarity measure between biocodes is based on the Hamming distance

Now we discuss the point of the binarization threshold choice. The value of the threshold τ_b is generally fixed at zero, owing to the theoretical and expected equal probability for each element (after random projection) to be positive or negative. This process is aimed at increasing the ambiguity for an attacker who wants to estimate a probable accepted biocode. But, instead of using the value 0, we propose to estimate the median over a training dataset A (in practice, the probability of a positive or negative element is not equal) and to set the threshold to the obtained value.

We suggest estimating this median from a fixed set of independent fingerprint images as in Fig.4:

For each attributed seed S Do
- For each image in the set A Do
  Compute $X = (x_1, x_2, \dots x_n)$ the biometric feature vector
  Generate the orthonormal matrix R_{nm} from the seed S.
  Compute $W(w_1, w_2, \dots w_m)$ with $w_i =< X|R_{ni} >, i = 1..m$
- Let g be the number of computed W.
- For i=1:m Do
  Compute M_i the median of the elements $(w_{ij}), j = 1..g$
- The vector$(M_1, M_2, \dots M_m)$ is considered as the threshold vector for the binarisation step.

Fig. 4. Median estimation protocol

Therefore, we obtain a protected MinuCode noted *PMinuCode* (*PMinuCode*$\in \{0,1\}^m$, *m the biocode length).* A protected template *PT* will be represented by the following notation:

$PT = \left\{PMinuCode_i + \{a_j\}_{j=1}^{K}\right\}_{i=1}^{v}$ while a_j denotes a minutia label.

Template matching: we have to develop a fingerprint matching algorithm in the transformation space. The algorithm receives as input two protected templates and delivers a matching score that expresses the degree of similarity between them. Let PT_A be the enrolled template containing *N* minutiae and PT_B the input template containing *M* minutiae obtained after registration of the image B relatively to the orientation of the image A as explained in Fig.2.

We need to identify a set of corresponding minutiae pairs: $C = \{(a_i, b_j) \,/\, a_i \in PT_A \; and \; b_j \in PT_B\}$ with the following minutiae pairing algorithm. Intuitively, two minutiae are paired if they satisfy local and global constraints. *i).* A local constraint is satisfied by the pair (a_i, b_j) if the Hamming distance $D(a_i, b_j)$ between their *PMinuCodes* is sufficiently small relative to a certain training measure. *ii).* A global constraint is satisfied if the minutia a_i is geometrically close to the minutia b_j (the K-plet structure is used to manage this constraint).

Now, the matching algorithm is divided into three phases:

***Phase1*:** it consists of the selection of the best matched pair {root1, root2}, root1$\in PT_A$ and root2$\in PT_B$ as following: Two minutiae a_i, b_j are likely to be paired if the distance between their *PMinuCodes* $D(a_i, b_j)$ is small. However, because of the possible overlap between ROIs of neighboring minutiae, a_i can also reveal a low distance with another minutia $b_{j'} \in PT_B$ different from b_j. Thus, to find the most distinguishable minutia pair, we try to minimize the following probability as in [25]:

$$P(a_i, b_j) = D(a_i, b_j)^2 / \left(\sum_{i'=1}^{n} D(a_{i'}, b_j) + \sum_{j'=1}^{m} D(a_i, b_{j'}) - D(a_i, b_j)\right) \quad (2)$$

The pair (root1,root2) is validated only if $P(a_i, b_j) < Threshold$. If (root1,root2) is validated, go to phase 2. Otherwise, go to phase3.

***Phase2*:** consider root1, root2 the first nodes to be explored in PT_A, PT_B resp. Now, we have to match the K-plet of root1 with that of root2. Each matched pair will be pushed in a queue and marked as visited. This selection scheme will now be recursively repeated until the queue becomes empty (each time, we pop the new pair (root1,root2) from the queue head). To avoid the local minima problem, go to phase1 considering just non visited minutiae;

Phase3 (Matching score computation*)*: In automatic system, the number of pairing minutiae *PM* is converted into a similarity score for normalization. We find that the following score: $\frac{PM}{\min(M,N)}$ is more appropriate according to experiences we done.

IV. EXPERIMENTS

To comply with the majority of works done on template protection, we use the FVC2002-DB2 [17] database. It consists of 100 fingers with 8 impressions per finger obtained using an optical sensor of 569 dpi. The size of the images is 560×296. We put *B*=3 (number of bands in the tesselation) implying n=384 (MinuCode size) while the length of each biocode is maintained at m=180 bits.

Biometric Performance evaluation: this first set of experiments is meant to evaluate the registration process against the orientation deformation. The evaluation will be performed for different image types. We set $period = 16$.

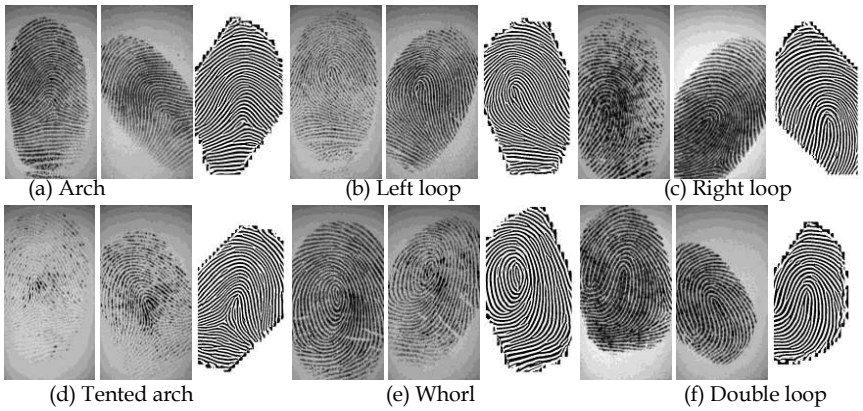


Fig. 5. Qualitative appreciation of the registration process (rotation handling) by image type. For each type, we have the enrolled, the input and the corrected images

As we remark in Fig.5, the correction works quite well. However, the main drawback of this method is that the estimation depends on the reference point. As a conclusion, we decide to rely on registration phase, knowing that it works even if it is not the best method.

Next, we evaluate the biometric performance verification of the approach concerning these three scenarios: the use of the sole biometric (SOLE), the protected template in both the Best (BEST) case when an impostor never steals the key, and in the worst (WORST) scenario when an impostor always steals the key.

In our experiments, the first impression of each finger is used as the enrolled fingerprint. The remaining 7 impressions are used as genuine queries. To compute the false acceptance rate, the first sample of each finger is matched against the first sample of the remaining fingers. We use the Half Total Error Rate ($HTER = \frac{FAR+FRR}{2}$) as a performance measure of the EER.

The performance of the sole biometric is reported in the form of receiver operating characteristic (ROC) curve as shown in Fig.6. The EER reported is 4.56%.

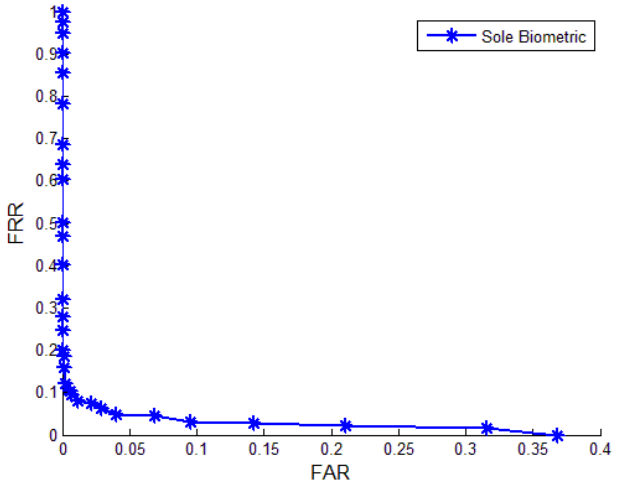


Fig.6. ROC curve of the sole biometric

Considering the protected biometric, we show in Fig.7 bellow the distribution of the False Rejection Rate (FRR) scores against the False Acceptance Rate (FAR) for all possible operating points. We plot both the best and worst cases on the same figure to analyze the impact of the decision threshold. The intersection between the FAR and FRR distributions represents the Equal Error Rate (EER).

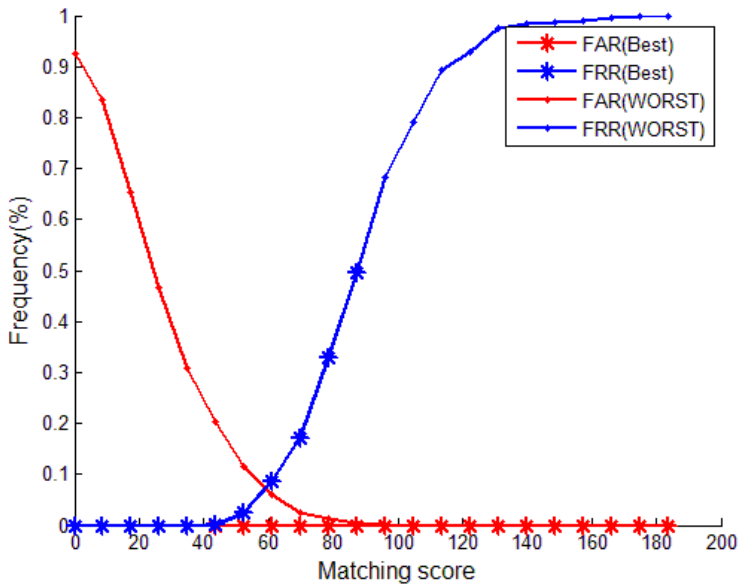


Fig. 7. Impostor vs. genuine distributions in both best and worst cases

From Fig.7, we observe that the BioHashing performance in term of EER is 0% in the best case and is equal to 6.68% for the worst case. However, if we pay a little attention to the scores distribution, we find that this assert is unrealistic in practice. In fact, in Fig. 7, we can see that these error rates are feasible, but using different decision thresholds: 40 for the best case and 60 for the worst one. However, It is inconceivable in a practical system to speak of different threshold values at the same time. A compromise on a single value should be done. If we tune the system at the same decision threshold, the real performance of BioHashing process will really fluctuate. If we adjust the system to get EER=0% under the assumption that the key is never stolen (as it is always suggested), the FAR in the worst case will be 22% which is exagerated for a real authentication system. It is then more appropriate to put the operating threshold to 60. Here, we find that the real performances of the system are: FAR=0 % in the never-stolen key case, FAR=7.16% when the key is always stolen and FRR=6.21% in both the scenarios which is normal because the knowledge or not of the key does not affect the genuine distribution.

V. SECURITY ANALYSIS

We make analysis in term of security and privacy vulnerabilities.

Threat evaluation in term of security: here, we evaluate the possibility of an intrusion success in the biometric system. Globally, we think that an impostor can conduct the following attacks:

Zero-effort attack: in this scenario, the impostor tries to impersonate the genuine user with unknown key by presenting its biometric. This attack will always fail because the FAR equals to zero.

Stolen-key attack: in this scenario, the impostor tries to impersonate the genuine user with available key but by presenting its proper biometric. This attack depends on the FAR which is 7.16%.

Brute-force attack: in this scenario, the impostor decides to overcome the feature protection component by sending a ready template to the matcher. He will try to estimate an accepted template by an exhaustive search. Let k the number of minutiae neighbours. It is sufficient for the impostor to

estimate (k+1) MinuCodes (equivalent to one k-plet). The complexity of such estimation is $2^{mx(k+1)}$. It is 8820 bits.

Attack by template correlation: the Brute-force attack being impossible, the attacker will try to predict an accepted template after eavesdropping *N* templates of the genuine user. We will now explore if a prediction of the $N+1^{th}$ template is possible. Because the template is a set of binary strings, we can study the correlation by Hamming distance distribution as done by Daugman in [20]. For this, we compare the Hamming distance distributions of minutia biocodes of each user generated with the same key (genuine distribution) and the same minutiae set but diversified from N keys (Pseudo-imposter distribution). We observe in Fig.8 that both distributions are not overlapped. The mean of the blue zone is 33%. Let *X* be the random variable that represents the number of no matched bits in the biocode. The pseudo-imposter distribution can then be approximated by a binomial distribution where the binary event is the fact that two bits are equal or not and the number of trials is *m*, the biocode length. Then the mean of this distribution is $m \times P$ with *P* the probability of the binary event. If we consider $Y = \frac{X}{m}$, the random variable of the normalized hamming distance then *P* will be equal to the mean 33%. This informs that the probability of predicting a correct value by bit is 1-*P*=67%, ideally we would get 50% for satisfying a total ambiguity. However, since the distributions of the genuine user and that of the pseudo-impostor are sufficiently separated, obtaining 67% is not negative.

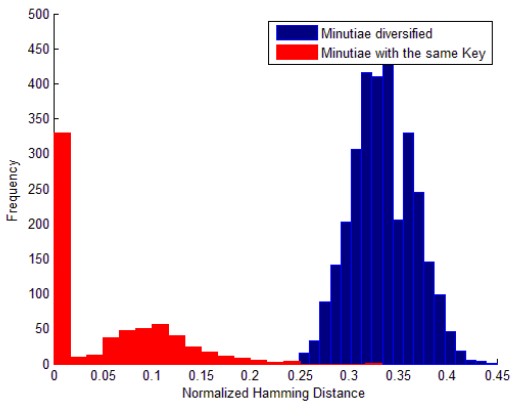


Fig. 8. Genuine vs. pseudo-impostor distributions by minutiae for N=10

Threat evaluation in term of privacy: we evaluate privacy in term of possibility of linkage and irreversibility of the protected template.

Irreversibility: we first assume that an attacker has access to both the random matrix R(n,m) and (k+1) PMinuCodes $(y_1, y_2, \dots, y_{k+1})/y_i \in \{0,1\}^m$, $i = 1..k+1$ corresponding to one k-plet and wants to estimate the original MinuCodes $(x_1, x_2, \dots, x_{k+1})/x_i \in \mathbb{R}^n$. From a mathematical view point, the invertibility question can be considered as finding a solution to the following linear equations system:

$$\begin{cases} R.x_1 = y_1 \\ \quad \vdots \\ R.x_{k+1} = y_{k+1} \end{cases}$$ with $(k+1)n$ unknowns and $(k+1)m$ equations,$(m < n)$. In this system, we have always more unknowns than independent linear equations. Based on the premise that possible solutions are infinite, the random projection is considered as irreversible. However, if an attacker knows the linkage among *I* diversified templates from *I* different matrices $R_i, i = 1..I$ and if Rank$[R_1, \dots R_I] = n$, the problem will be invertible (for n=384, m=180, *I* needs to be equal to 3). Fortunately, this attack remains difficult because finding 3 related diversified templates in our system is difficult.

Unlinkability/Diversity: a common way to evaluate diversity is to match different transformed templates obtained from the same biometrics after assigning each user *t* different keys. We call the

percentage of success match the cross matching rate. For the proposed model, the matching score between diversified templates is always 0% for any value of *t* (we vary *t* from 2 to 10). It is completely independent from any operating threshold. This is a very interesting property if the same algorithm is used for different systems tuned at different operating thresholds.

VI. A COMPARATIVE STUDY

We intend now to compare in Table I the proposed method with ones dealing with minutiae features in order to situate our contribution. The EER is at each time in the worst case.

TABLE I PERFORMANCE COMPARISON BETWEEN EXISTENT MINUTIAE PROTECTION SCHEMES WHEN ALL ALGORITHM PARAMETERS ARE KNOWN

Methods	Database	EER	Diversity
Proposed method	FVC2002-DB2	6.68%	Good and threshold independent (good diversity means no match is found between diversified templates)
Ang et al. [27]	NIST (80 images)	16.8%	70% chance of correctly linking diversified templates
Kumar et al. [19]	FVC2002-DB2	4.98%	Not measured
Farooq et al. [18]	1 enrolled sample, 1 enrolled test for 1000 users.	1.59%	Good but threshold dependent
Lee et al. [15,16]	FVC2004-DB2	10.3%	Good but threshold dependent
Yang et al. [14]	FVC2002-DB2	5%	Good but threshold dependent
Boult et al. [30]	FVC2002-DB2	Not reported	Not reported
Ratha et al. [24]	1 enrolled sample, 1 enrolled test for 188 users.	10%	91.5% chance for correctly linking diversified templates.
Nanni et al. [36]	FVC2002-DB2	3.45%	Not studied

From Table I, we remark that the proposed method is well situated. Comparing to the method of Yang *et al.*, it presents better diversity but slightly worst performance. This is due to the nature of the descriptor. Indeed, as detailed before, we use minutia texture descriptor where the relation between minutiae is just expressed by neighborhood. The method proposed by Yang expresses the same relation resorting to geometric information (minutiae vicinity). As a perspective, we intend to extend the neighborhood with more information and to fuse the texture descriptor with the Tico descriptor [25] used by Nanni *et al.* [42] which seems more discriminative. Effectively Tico report-

ed on the FVC2002-DB2 an EER=2.3% for the sole biometric when our descriptor gives 4.56%. The strength of our proposal is then in the diversity property where the cross match rate is always 0%, independently from any decision threshold.

VII. CONCLUSION

We discuss in this paper the security and privacy concerns over the use of biometrics. A focus has then been made on the dual factor BioHashing transformation for its diversity property comparing to other methods. We highlight the importance of using minutiae and propose an adaptation of BioHashing to minutiae templates. However, we experimentally demonstrate that the foremost claim of having near zero equal error rate when nobody steals the key factor is not exact in practice and if this is kept, the system will have a great intrusion risk. The accuracy achieved in the worst scenario where all the system parameters are known is 6.68% and 3.10% when no key is revealed, it is 4.56% for the sole biometric. So the trade-off between security and privacy is confirmed but it is not so drastic. In term of privacy preservation, we show that diversity is achieved in 100% of cases independently of any operating threshold while just a little correlation between diversified templates was quantified. We also show the strength of the approach against different scenario attacks. We plan to extend this work in several directions especially with the local structure between minutiae which we intend to enrich with additional geometric information.

VIII. REFERENCES

[1] A.K. Jain, K. Nandakumar, and A. Nagar, 'Biometric template security,' EURASIP J on Advances in Signal Processing, vol. 12, 2008.

[2] A.K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, 'Filterbank based fingerprint matching,' IEEE Trans Image Process, vol. 9, no. 5, 2000.

[3] A. Juels and M. Sudan, 'A Fuzzy Vault Scheme,' Proc. of IEEE International Symposium on Information Theory, 2002.

[4] A. Juels and M. Wattenberg, 'A Fuzzy Commitment Scheme,' 6th ACM Conference on Computer and Communications Security, 1999.

[5] A. Kong, K.H. Cheung, D. Zhang, M. Kamel, and J. You, 'An analysis of BioHashing and its variants,' Pattern Recognition, vol.

39, 2005.

[6] A. Lumini and L. Nanni, 'An improved BioHashing for human authentication,' Pattern recognition, vol. 40, pp. 1057-1065, 2007.

[7] A. Nagar, K. Nandakumar, and A.K. Jain, 'Biometric template transformation: A security analysis,' Proceeding of SPIE, 2010.

[8] A. Nagar, K. Nandakumar, and A.K. Jain, 'A hybrid biometric cryptosystem for securing fingerprint minutiae templates,' Pattern Recognition Letters, vol. 31, no. 8, pp. 733-741, 2009.

[9] A.B.J. Teoh, Y.W. Kuan and T. Kar-Ann, 'Cancellable biometrics and user-dependent multi-state discretization in BioHash,' Pattern analysis & applications, vol. 13, pp. 301-307, 2010.

[10] A.B.J. Teoh, Y. Kuan and S. Lee, 'Cancellable biometrics and annotations on BioHash,' Pattern recognition, 2007, pp. 2034-2044.

[11] A.B.J. Teoh, B. Jin, T. Connie, D. Ngo, and C. Ling, 'Remarks on BioHash and its mathematical foundation,' Information processing letters, vol. 100, no. 4, pp. 145-150, 2006.

[12] A.B.J. Teoh, D. Ngo, and A. Goh, 'BioHashing: two factor authentication featuring fingerprint data and tokenised random number,' Pattern recognition, vol. 37, no. 11, 2004.

[13] A.B.J. Teoh, D. Ngo, and A. Goh, 'An integrated dual factor authenticator based on the face data and tokenised random number,' 1st International conference on biometric authentication (ICBA), 2004.

[14] B. Yang, D. Hartung, K. Simoens, and C. Busch, 'Dynamic random projection for biometric tempalte protection,' 4th IEEE BTAS, 2010.

[15] C. Lee and J. Kim, 'Cancelable fingerprint templates using minutiae-based bit-strings,' J of Network and Computer Apps, vol. 33, 2010.

[16] C. Lee, J. Choi, K.A. Toh, S. Lee, and J. Kim, 'Alignment-Free Cancelable Fingerprint Templates Based on Local Minutiae Information,' IEEE trans on systems, man, and cybernetics, 2007.

[17] D.Maio, D. Maltoni, J. L. Wayman, and A. K. Jain, 'FVC2002: Second Fingerprint Verification Competition,' ICPR, Canada, 2002.

[18] F. Farooq, R.M. Bolle, J. Tsai-Yang, and N. Ratha, 'Anonymous and Revocable Fingerprint Recognition,' IEEE Conf on CVPR, 2007.

[19] G. Kumar, S. Tulyakov, and V. Govindaraju, 'Combination of Symmetric Hash Functions for Secure Fingerprint Matching,' 20th Int conf on pattern recognition (ICPR), 2010.

[20] J. Daugman, 'The importance of being random: statistical principles of iris recognition,' Pattern Recognition, vol. 36, no. 2, pp. 279-291, 2003.

[21] K. Nandakumar, A. Nagar, and A.K. Jain, 'Hardening fingeprint fuzzy vault using password,' International conference on biometrics, 2007.

[22] J.M.G. Linnartz and P. Tuyls, 'New shielding functions to enhance privacy and prevent misuse of biometric templates,' 4th Inter Conf on Audio- and Video-Based Biometric Person Authentication, 2003.

[23] M. Liu, X. Jiang, and A.C. Kot, 'Fingerprint reference-point detection,' EURASIP J on Applied Signal Processing, vol. 4, 2005.

[24] M. Tico and P. Kuosmanen, 'Fingerprint Matching Using an Orientation-Based Minutia Descriptor,' IEEE Trans on PAMI, 2003.

[25] N.K. Ratha, S. Chikkerur, J.H. Connell, and R.M. Bolle, 'Generating cancelable fingerprint templates,' IEEE Trans on PAMI, 2007.

[26] N.K. Ratha, J.H. Connell, and R.M. Bolle, 'Enhancing security and privacy in biometrics-based authentication systems,' IBM Systems Journal, vol. 40, no. 03, pp. 614–634, 2001.

[27] R. Ang, R. Safavi-Naini, and L. McAven, 'Cancellable key-based fingerprint templates,' ACISP, pp. 242-252, 2005.

[28] S. Chikkerur, A.N. Cartwright, and V. Govindaraju, 'K-plet and Coupled BFS: A Graph Based Fingerprint Representation and Matching Algorithm,' Proc. Int. Conf. on Biometrics, pp. 309–315, 2006.

[29] S. Kaski, 'Dimensionality reduction by random mapping,' Int. Joint Conf. on Neural Networks, pp. 413-418, 1998.

[30] T. Boult, W. Scheirer, and R.Woodworth, 'Revocable fingerprint biotokens: accuracy and security analysis' Int Conf CVPR, 2007.

[31] W. Zhang and Y. Wang, 'Core-Based Structure Matching Algorithm of Fingerprint Verification,' 16th Int Conf on Pattern Recognition, 2002.

[32] Y. Dodis, L. Reyzin, and A. Smith, 'Fuzzy extractors: How to generate strong keys from biometrics and other noisy data,' In Adv. in Cryptology Eurocrypt, pp. 523-540, 2004.

[33] Y. Sutcu, S. Rane, J. Yedidia, S. Draper, and A. Vetro, 'Feature extraction for a slepian-wolf biometric system using ldpc codes,' Proc of the IEEE Int Symposium on Information Theory, 2008.

[34] R. Belguechi, C. Rosenberger, S. Ait-Aoudia, 'BioHashing for Securing Minutiae Template', 20th International Conference on Pattern Recognition (ICPR), 2010.

[35] N.Radha, and S.Karthikeyan, 'An Evaluation Of Fingerprint Security Using NonInvertible Biohash', International Journal of Network Security & Its Applications (IJNSA), vol. 3, 2011.

[36] L. Nanni, S. Brahnam, A. Lumini, 'Biohashing applied to an orientation-based minutia descriptor for a secure fingerprint authentication system', 2011.

[37] W. Scheirer and T. Boult. 'Cracking fuzzy vaults and biometric encryption'. Proc. of Biometrics Symposium, 2007.

[38] A. Kholmatov, B. Yanikoglu. 'Realization of correlation attack against the fuzzy vault scheme'. Proceedings of SPIE: Biometrics, Security, Forensics, steganography and Watermarking of Multimedia Contents, San Jose, USA, 2008.

[39] A. Lumini, L. Nanni. 'Empirical Tests on BioHashing'. NeuroComputing (69:16), 2006, pp. 2390-2395.

[40] A. Kanak, I. Sogukpinar. 'Classification Based Revocable Biometric Identity Code Generation'. BioID MultiComm, 2009.

[41] L. Nanni, A. Lumini. 'Local binary patterns for a hybrid fingerprint matcher'. Pattern Recognition Journal (41), 2008.