A Locality Sensitive Hashing Based Approach for Generating Cancelable Fingerprints Templates

Debanjan Sadhya
ABV-IIITM Gwalior
Gwalior, MP, India
debanjan@iiitm.ac.in

Zahid Akhtar
University of Memphis
Memphis, TN, USA
zmomin@memphis.edu

Dipankar Dasgupta
University of Memphis
Memphis, TN, USA
ddasgupt@memphis.edu

Abstract

Cancelable biometric schemes have emerged as a promising approach for providing robust security guarantees to extracted biometric features. In this paper, we develop a working framework for generating secure templates from raw fingerprint images utilizing the notion of Locality Sampled Codes (LSC). For achieving such objectives, we initially represent the features of a fingerprint image in a binarized form, and subsequently generate the final cancelable template by sampling random bit locations from it. Since the LSC technique is functionally established on the principle of Locality Sensitive Hashing (LSH), the induced transformations do not degrade the performance of the overall biometric model. We have performed a thorough theoretical analysis coupled with comprehensive empirical justifications for investigating the fulfillment of properties like non-invertibility, revocability, and unlinkability. We have also analyzed the performance of the model over the FVC2002-DB1, FVC2002-DB3, FVC2004-DB1, and FVC2004-DB3 fingerprint databases, for which we have obtained comparatively low EERs of 0.19%, 1.44%, 1.28% and 2.72% respectively in the stolen-token scenario.

1. Introduction

Biometrics has become a ubiquitous technology for establishing access control mechanisms via the process of identity authentication. These pattern recognition frameworks have been widely accepted by both government and semi-government organizations for large-scale deployments over a variety of domains. However, the privacy and security aspects associated with such models is still a matter of prime concern since an active adversary can easily exploit their structural and functional vulnerabilities. Some canonical instances of adversarial attacks on biometric frameworks include privacy invasion, hill climbing, masquerade attack, replay attack, and identity theft [8].

978-1-7281-1522-1/19/$31.00 ©2019 IEEE

Biometric template protection (BTP) schemes are designed to provide provable guarantees against biometric security threats. Based upon their structures, BTP schemes are classified as either cancelable biometrics or biometric cryptosystems [21]. This paper deals with the former notion, in which the original biometric features are transformed into protected templates via the use of one-way transformation functions [18]. A successfully designed cancelable biometric model should fulfil the essential characteristics of *Non-invertibility*, *Unlinkability*, *Revocability*, and *Performance* [21, 18]. Non-invertibility refers to the property that a protected template cannot get easily inverted into its corresponding base feature. Unlinkability states that multiple transformed templates which originate from the same base feature should be virtually indistinguishable. The revocability guarantee explores the possibility of generating new and distinct features in case of a database breach. Finally, the property of performance states that the recognition accuracy of the cancelable model should not degrade substantially from that of the baseline model.

The study in this paper proposes a generic framework for generating cancelable templates from binary fingerprint features. The core of work pivots around the Locality Sensitive Hashing (LSH) technique, which maps input data in a specific manner so that similar data map to the same localities with high probability. Our work is an extension of the study of Sadhya and Raman [22], wherein the authors have generated cancelable Locality Sampled Codes (LSCs) from iris features via utilizing a randomized sampling strategy. In our work, we initially extract Minutia Cylinder-Codes (MCCs) [1] from fingerprint images, and subsequently binarize them via a zero-thresholding based technique. Finally, we generate the LSC features from these binarized vectors for creating the cancelable templates. The main advantage of our proposed work is the simultaneous fulfillment of strong security guarantees and non-degradation of the system performance (in both the genuine and stolen token scenarios).

Since our developed model is based on the LSH principle, it also satisfies the properties of non-invertibility, unlinkability and revocability. Hence, our study not only successfully investigates the applicability of the LSC based scheme for fingerprints but also provides a holistic framework for efficiently generating cancelable fingerprint templates.

The rest of the paper is organized as follows. Section 2 discusses some related studies, and Section 3 presents some preliminary notions associated with our work. Section 4 forms the crux of our work where we present the framework for generating cancelable fingerprint templates. Section 5 presents several relevant empirical results, and Section 6 analyzes the security guarantees of our model. Finally, we conclude our work in Section 7.

2. Related Work

We initiate this section by discussing several techniques which generate cancelable templates utilizing non-invertible transforms. Although the core concept was developed by Ratha *et al*. [20] long back in 2007, this design was successfully cracked later by Quan *et al*. [19]. Another pioneering approach for generating cancelable fingerprint templates was the BioHashing scheme [26]. In this technique, unique binary code was created based on iterated inner products between tokenized pseudo-random numbers and individual fingerprint features. However, some later works [4, 17] refuted the non-invertibility property of BioHashing if the stored protected templates get compromised. Hash functions are another popular primitive which provide provable non-invertibility and diversity guarantees. This security notion was initially utilized by Tulyakov *et al*. [27], and later extended by Sadhya and Singh [23] for generating cancelable fingerprint tokens.

Pre-alignment is an undesirable factor in the design of fingerprint-based cancelable biometric models. Alignment-free techniques are comparatively more efficient both in terms of computational complexity and recognition accuracy rates. The notion of Curtailed circular convolution was used by Wang and Hu [30] for designing such a scheme. The security of this technique is based on the one-way transformation functions which map input binary strings to convolved output vectors. Lee *et al*. [15] used information about the orientation field around a minutiae point for constructing a translation invariant scheme. However, it demonstrates poor results for low-quality fingerprints. Other related works include that of Wang and Hu [29] wherein a densely infinite-to-one mapping (DITOM) technique was used for pairing minutiae vectors and of Yang *et al*. [32] who exploiting the local Voronoi neighbor structures (VNSs) of the fingerprints. More recently, the discriminating capability of zoned minutia pairs was used by Wang *et al*. [31] for improving the recognition performance of the overall model.

Finally, we discuss some cancelable schemes which generate the protected data in the form of binary feature descriptors. The first attempt for creating anonymous and revocable representation from binary fingerprint features was made by Farooq *et al*. [5]. However, this technique requires greater computational resources due to the estimation of all possible invariant features. Lee and Kim [16] proposed a framework which mapped the input minutiae into a predefined 3-D array. However, the stored cancelable data becomes vulnerable to adversarial attacks in the stolen token scenario. This idea was later utilized by Jin *et al*. [12], who proposed a polar grid based 3-tuple quantization (PGTQ) scheme. Most recently, the partial Hadamard transformation was used to secure binary fingerprint representations [28]. This work reported very satisfactory performance metrics since the transformation function preserves the stochastic distance between the binary features. The notion of Locality Sensitive Hashing (LSH) was used by Jin *et al*. [9] to develop the *Index-of-Max* (IoM) hashing scheme. This technique transforms real-valued fingerprint features into discrete indexed hashed codes. Although our present work also employs the notion of LSH, its realization for generating the protected templates is entirely different. In contrast to the IoM scheme, we have randomly sampled bits from binary fingerprint feature vectors for constructing the local hashes of the LSH.

3. Locality Sensitive Hashing (LSH)

The LSH procedure maps similar input features into the same location with very high probability. Since LSH is essentially a hashing technique, the size of the input features is much larger than the number of locations. The LSH family is theoretically described as:

Definition 1 (LSH [3]) *A LSH is a probability distribution on a family of $\mathcal{H}$ of hash functions h such that $\mathbb{P}_{h\in\mathcal{H}}[h(X) = h(Y)] = d(X, Y)$ where d is a distance function defined on the collection of object X and Y.*

The core of the LSH scheme comprises of several local hash functions (h_i). The utilization of these functions enables in an accurate estimation of the pairwise distance of the input items in the hashed domain. In this way, LSH ascertains that the collision probability between the hashes of two similar items remains comparatively higher, and vice-versa. Theoretically stating,

$$\begin{aligned} &\mathbb{P}_{h\in\mathcal{H}}(h_i(X) = h_i(Y)) \geq P_1, \quad \text{if} \quad d(X, Y) < R \\ &\mathbb{P}_{h\in\mathcal{H}}(h_i(X) = h_i(Y)) \leq P_2, \quad \text{if} \quad d(X, Y) > cR \\ &\text{where } P_1 > P_2. \end{aligned} \tag{1}$$

In Eqn. (1), P_1 and P_2 are two probabilistic collision bounds of h_i, $X, Y \in \mathbb{R}^d$, $\mathcal{H} = \{h : \mathbb{R}^d \rightarrow U\}$, $d(.,.)$ is the

distance function and U is the hashed space. Furthermore, R is the distance threshold and c is an approximation factor which is always greater than 1. This notion of LSH was recently used by Sadhya and Raman [22] for developing cancelable templates from iris features (commonly termed as IrisCode). In their work, the authors randomly sampled bits from these feature IrisCodes for generating the local hashes.

4. Methodology

Our proposed cancelable framework comprises of four distinct modules: feature extraction, pre-processing, LSC generation and matching. Every individual module is described in details as follows. A schematic diagram illustrating the entire procedure of generating LSC templates from raw fingerprint images is presented in Fig. 1.

4.1. Feature Extraction

The first stage of any biometric model involves the extraction of unique and discriminating features from the corresponding raw images. In our work, we have extracted Minutia Cylinder-Codes (MCCs) [1, 2] from the input fingerprint images. MCC is a state-of-the-art local minutiae descriptor which constructs 3D cylindrical structures by encoding spatial and directional information between a minutiae and its neighborhood of radius R. This technique creates a cylinder set CS from a minutiae template T, which contains the ISO/IEC 19794-2 standard triplet encoding for each minutia point m. Each cylinder in the set has the advantages of being invariant to translational and rotational variations. Furthermore, the MCC representation tolerates missing and spurious minutiae since it is based on a fixed-radius approach. For more details, the readers are referred to [1].

The MCC descriptors provide excellent system performance in terms of recognition accuracy. However, they are susceptible to many adversarial attack forms since they leak information about the position and angle of the minutiae points. Although this problem was later addressed by introducing non-invertible Protected Minutia Cylinder Code (P-MCC) [6], it did not provide the guarantees of diversity and unlinkability. In our work, we have analyzed all the security aspects of our scheme both empirically and theoretically.

4.2. Pre-processing

During the pre-processing phase, we convert the MCC fingerprint descriptor into a fixed-length binary representation. We require this form of feature representation since the protected LSC templates can be generated from only binary feature vectors. For achieving this purpose, we initially convert the MCC descriptor into a fixed-length real-valued format via the Kernel Principal Component Analysis (KPCA) framework [10] . As suggested by the name itself, the crux of this scheme utilizes a KPCA based projection matrix. KPCA is a generalized version of Principal Component Analysis (PCA), which does not consider the assumption that the input data is multivariate Gaussian distributed [25]. The fundamental idea behind KPCA is the application of a nonlinear function $\Phi(.)$ to the data points $\mathbf{x}_j$ on the kernel space, $\mathbf{x}_j \in \mathbb{R} \mid j = \{1, 2, \ldots, l\}$. Based upon these non-trivial functions, a kernel matrix $\mathbf{K}$ of dimension $l \times l$ is defined as [10]:

$$\mathbf{K} = \langle \Phi(x).\Phi(y) \rangle = \Phi(x)^T \Phi(y) \tag{2}$$

where (x, y) are points in the kernel space, and T is the matrix transpose operator. In the KPCA framework, the fingerprint descriptors are initially partitioned into non-overlapping training and testing set. A kernel matrix is then constructed by using the following kernel function:

$$\mathbf{K}(i,j) = \exp\Big\{ -(1 - \mathbb{S}(i,j))^2 / 2\sigma^2 \Big\}. \tag{3}$$

In Eqn. (3), $\mathbb{S}$ is the dissimilarity measure between two MCC based minutiae representations, and σ is the spread factor of the kernel. Subsequently, the projection matrix $(\overline{\boldsymbol{\alpha}})$ is computed by estimating the eigenvectors of this kernel. Furthermore, a fixed feature vector $(\mathbf{v}^{fl})$ is calculated by concatenating all the similarity scores obtained from matching the MCC descriptors of the testing samples with all the remaining training samples. A transformed feature vector $(\overline{\mathbf{v}}^{fl})$ is next generated from $\mathbf{v}^{fl}$ by using the kernel function defined in Eqn. (3). The final fixed-length feature representation $(\mathcal{T})$ is formed by projecting $\overline{\mathbf{v}}^{fl}$ onto $\overline{\boldsymbol{\alpha}}$. Thus,

$$\mathcal{T} = \overline{\mathbf{v}}^{fl}.\overline{\boldsymbol{\alpha}} \tag{4}$$

Interestingly, the length of $\mathcal{T}$ depends upon the number of desired output dimensions, which is further determined by the number of training samples. In our work, we binarize $\mathcal{T}$ by utilizing a basic zero-thresholding based quantization scheme. We denote this fixed-length binary template by $\mathcal{T}_B$. The rule governing this quantification technique can be stated as:

$$b_i = \begin{cases} 1 \text{ if } \mathcal{T}_i \geq 0 \\ 0 \text{ if } \mathcal{T}_i < 0 \end{cases} \tag{5}$$

where, b_i is the i^{th} bit which is assigned to the binary feature vector $\mathcal{T}_B$, and $\mathcal{T}_i$ is the corresponding element in $\mathcal{T}$. This process completes our pre-processing phase.

Code available at: `https://sites.google.com/site/jinzhe/about-me`

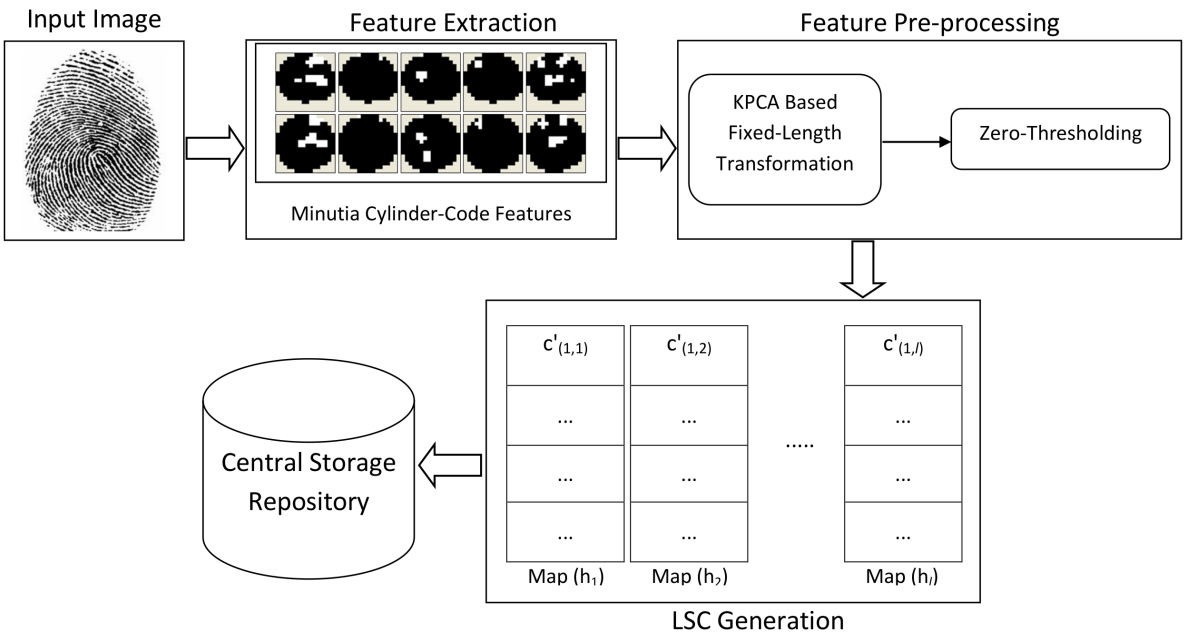


Figure 1. The proposed framework for generating cancelable LSC features from raw fingerprint images.

4.3. LSC Generation

In our proposed model, cancelable LSC features are generated from the binarized feature codes through the process described in [22]. We initially divide $\mathcal{T}_B$ into n non-overlapping blocks, which is represented by the set $\mathcal{B}$. For every block $B_i \mid B_i \in \mathcal{B}$, k indices are randomly sampled from it for generating the set of l independent hash functions. Hence this process essentially translates to extracting a random permutation sequence of indices from each block. An individual hash function is represented by h_j, and their set by $\mathcal{H}(B_i)$. All the indices which are sampled by $\mathcal{H}(B_i)$ are collectively termed as *marked positions* (denoted by $[B]$), and the rest of them as *unmarked positions*. Subsequently, the actual bits in the marked positions for block B_i are extracted and their decimal equivalents are computed (termed as *block components*). For a block B_i, an individual block component is denoted by $c_{(i,j)}$. This step is followed by the application of a modulo function for ensuring the guarantee of non-invertibility. The output of the modulo function is controlled by a security parameter T, which lies in the range $[0, 1]$. This encoding process is formally represented as:

$$c'_{(i,j)} = c_{(i,j)} \bmod \lfloor T \times 2^k \rfloor \tag{6}$$

where, $c'_{(i,j)}$ represents the corresponding non-invertible block components. Finally, these terms are stored in the form of maps (denoted by $Map(h_j)$, $j = \{1, 2...l\}$) in the data repository. The set of all the l maps collectively form the LSC feature for the fingerprint code.

4.4. Comparison and Score Generation

For any biometric model, a matching phase is required during which the feature descriptors of a probe biometric is matched against that of a reference biometric. In our case, this process pertains to matching the two LSC features which are obtained from a probe and reference fingerprint sample. A similarity score is generated in this process, which is subsequently utilized for making the final decision. This operation in our framework is carried out by block-wise estimating the average number of matching elements which are present in the individual maps of each LSC. The final similarity score (denoted by $\mathcal{S}$) is computed in the range $[0, 1]$ by estimating the average value over all the blocks. Hence,

$$\mathcal{S} = \frac{\sum_{i=1}^{n} \left(\frac{|e_j = q_j|}{l} \right)}{n} \mid j = \{1, 2...l\},\ \forall e_j \in E_i,\ \forall q_j \in Q_i \tag{7}$$

where E_i and Q_i denotes the set of non-invertible block components in the enrolled and probe LSCs respectively.

5. Empirical Results and Analysis

In our experimental section, we perform several relevant experiments for investigating the performance of our cancelable model. However, we initiate this section by discussing the databases used in our simulation, the comparison protocol, and the evaluation metrics.

Database	Sensor Type	Dimension	Resolution
FVC2002-DB1	Optical	388×374	500 dpi
FVC2002-DB3	Capacitive	300×300	500 dpi
FVC2004-DB1	Optical	640×480	500 dpi
FVC2004-DB3	Thermal	300×480	512 dpi

Table 1. Characteristics of the utilized fingerprint databases.

5.1. Simulation Setup

All of our results were validated on four benchmark fingerprint databases: FVC2002-DB1, FVC-2002-DB3, FVC2004-DB1, and FVC2004-DB3. Each of these databases contains eight fingerprint samples from 100 users, thereby totaling to 800 (100×8) images. For extracting the minutiae points from these databases, we used the data available in the Fingerprint Manual Minutiae Marker (FM3) toolkit [13]. FM3 is a GUI based tool which manually marks and extracts minutia features from fingerprint images in the standard ISO/IEC 19794-2 format. The salient characteristics of these databases are presented in Table 1.

We have used the standard FVC protocol for estimating the genuine and impostor scores. In this protocol, the False rejection rate (FRR) is calculated by matching every intra-class sample with each other, whereas the False acceptance rate (FAR) is computed by matching every first sample of a class with the first samples of the remaining classes. This matching procedure yields $\frac{5\times4}{2} \times 100 = 1000$ genuine scores and $\frac{100\times99}{2} = 4950$ impostor scores. The quantitative metrics which we have utilized for analyzing the performance of our proposed model include the Equal error rate (EER), decidability index (d'), and Detection Error Tradeoff (DET) curves (for some specific cases).

5.2. Performance Evaluation

We analyze the performance of our framework under different settings of the model parameters. We specifically study the effects of the number of blocks (n), the size of a hash function (k), and the security threshold (T). We would use the optimum values of these system parameters in our subsequent empirical works. It is noticeable in this regard that we do not vary the number of local hash functions (l) since it does not affect the overall system performance [22]. As such, we have fixed an average value of $l = 200$ throughout our simulation.

5.2.1 Effects of # of Blocks

We performed several experiments by varying $n = \{3, 6, 12\}$. The values of the associated block size corresponded to $b = \{100, 50, 25\}$ bits since the pre-processing phase resulted in the generation of binary feature vectors of size 300 bits. Noticeably, we did not increase the value of n more than 12 since the block size would consequently become very low. We have computed our results while fixing the average values of $k = 10$, and $T = 0.25$. The EER(%) and d' values for all the ensuing settings are shown in Table 2, wherein the performance of the model noticeably decreases with the block-size. The lowest EERs = $\{0, 0.41, 0.23, 0.73\}$ were all obtained for $n = 12$ corresponding to the four fingerprint databases respectively. We would fix this particular value of n in our ensuing experimental sections.

Database	# of Blocks (n)					
	3		6		12	
	EER (%)	d'	EER (%)	d'	EER (%)	d'
FVC2002-DB1	0	3.61	0	3.79	0	3.81
FVC2002-DB3	1.07	2.94	0.48	3.01	0.41	3.01
FVC2004-DB1	0.92	2.71	0.45	2.92	0.23	2.98
FVC2004-DB3	1.26	2.59	0.98	2.72	0.73	2.73

Table 2. Variation of EER (%) and d' with the number of blocks for fixed values of $k = 10$ and $T = 0.25$.

5.2.2 Effects of Hash Function Size

In our next experiment, we analyze the effects of the size of the hash functions (k) on the model performance. In accordance with the previous works, we varied $k = \{5, 10, 15\}$ for each of the fingerprint databases. We have not considered k<5 since decreasing the value of k increases the adversarial success probability in inverting the protected LSC templates. Furthermore, we have fixed the optimum value of $n = 12$ and the average value of $T = 0.5$ thorough this section. The EER(%) and d' for all the resulting scenarios are presented in Table 3, wherein it is observable that the lowest error rates correspond to $k = 10$ for FVC2002-DB3, and $k = 5$ for the remaining fingerprint databases. The lowest EERs were noted to be 0, 0.15, 0.2, and 0.59 corresponding to the FVC2002-DB1, FVC2002-DB3, FVC2004-DB1 and FVC2002-DB3 databases respectively.

Database	Hash Function Size (k)					
	5		10		15	
	EER (%)	d'	EER (%)	d'	EER (%)	d'
FVC2002-DB1	0	5.513	0	3.43	0.01	2.18
FVC2002-DB3	0.2	4.41	0.15	2.73	1.64	1.75
FVC2004-DB1	0.2	3.92	0.3	2.75	1.37	1.86
FVC2004-DB3	0.59	3.74	1.16	2.5	2.53	1.71

Table 3. Variation of EER (%) and d' with the size of hash functions for fixed values of $n = 12$ and $T = 0.5$.

5.2.3 Effects of Security Threshold

The final parameter which we investigate with regards to the system performance is the security threshold (T). In this simulation, we altered $T = \{0.15, 0.25, 0.5, 0.75\}$ for the optimum values of n and k. As discussed previously

FVC-2002: http://bias.csr.unibo.it/fvc2002/
FVC-2004: http://bias.csr.unibo.it/fvc2004/

in Section 4.3, the values of T are selected in $[0, 1]$ so that the mapping between the invertible and non-invertible block components becomes a surjection. All the relevant results are presented in Table 4, wherein it can be observed that a particular value of T does not result in the best recognition rates for all the four databases. Instead, the lowest EER $= \{0, 0.15, 0.11, 0.48\}$ resulted from $T = \{0.25, 0.5, 0.75, 0.25\}$ for the four fingerprint databases respectively. It can also be noticed that the EERs are comparatively higher for the FVC2002 databases in relation to the FVC2004 databases (especially for FVC2004-DB3). This observation can be attributed to the fact that the fingerprint images in the later database are of poor quality.

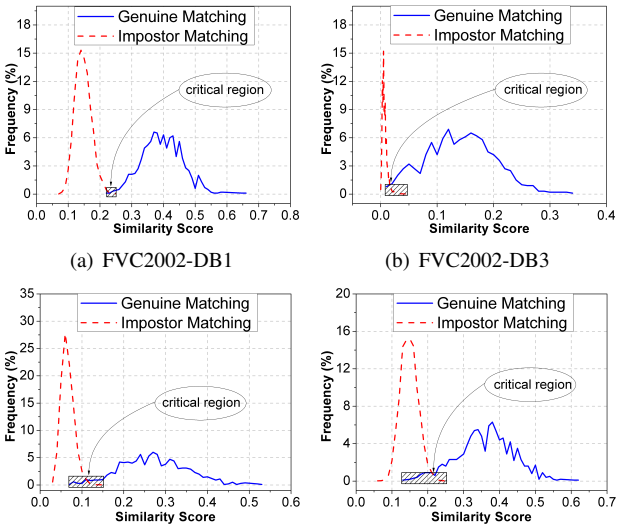
Database	Security Threshold (T)							
	0.15		0.25		0.5		0.75	
	EER (%)	d'	EER (%)	d'	EER (%)	d'	EER (%)	d'
FVC2002-DB1	0	5.65	0	5.69	0	5.51	0	5.15
FVC2002-DB3	1.86	2.42	0.41	3.01	0.15	2.73	0.35	2.52
FVC2004-DB1	0.22	4.24	0.35	4.09	0.2	3.92	0.11	3.68
FVC2004-DB3	0.5	4.08	0.48	3.92	0.59	3.74	0.5	3.51

Table 4. Variation of EER (%) and d' with the security threshold.

5.2.4 Stolen-Token Scenario

We also conducted experiments for evaluating the system performance under the stolen-token/lost-token scenario. In such cases, it is assumed that a user's key is stolen and subsequently used by the adversary. This scenario was imitated in our model by assigning the same key to all the database subjects. For our case, these keys correspond to the cancelable permutation tokens ($[B]$) which are stored in the database. The genuine scores were subsequently calculated by matching every sample within a class, whereas the impostor scores were estimated by matching the first sample of a class with the second sample of the remaining classes. Hence this protocol resulted in the generation of $\frac{5\times4}{2} \times 100 = 1000$ genuine scores and $\frac{100\times99}{2} = 4950$ impostor scores (similar to the FVC protocol).

The genuine and impostor score distributions under the stolen-token scenario are illustrated in Fig. 2. Furthermore, the supporting DET curves are also illustrated in Fig. 3. We have also compared the EER(%) and d' values under the two scenarios (for the optimized model parameters) in Table 5. In comparison to the traditional cases analyzed in Section 5.2, a marginal increment in the EER can be noticed for the stolen-token scenario. The highest EER = 2.72% was noted for FVC2004-DB3 since its enrolling subjects were advised to deliberately introduce some distortions during the data acquisition process.



(a) FVC2002-DB1 (b) FVC2002-DB3

(c) FVC2004-DB1 (d) FVC2004-DB3

Figure 2. Genuine and impostor score distributions of the proposed cancelable scheme under the stolen-token scenario.

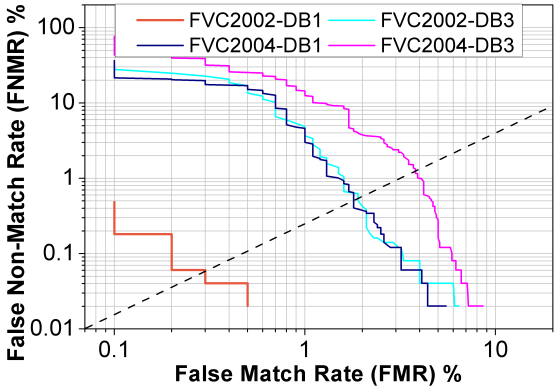


Figure 3. The DET curves of the four fingerprint databases under the stolen-token scenario.

Database	n	k	T	Genuine-Token		Stolen-Token	
				EER (%)	d'	EER (%)	d'
FVC2002-DB1	12	5	0.25	0	5.69	0.19	5.18
FVC2002-DB3	12	10	0.5	0.15	2.73	1.44	3.11
FVC2004-DB1	12	5	0.75	0.11	3.68	1.28	3.63
FVC2004-DB3	12	5	0.25	0.48	3.92	2.72	3.63

Table 5. Performance comparison under the genuine-token and stolen-token scenarios for the optimum model parameters.

5.3. Comparative Analysis

Finally, we compare the worst-case performance of our proposed model with other related state-of-the-art works. We report the stolen-token scenario EER (%) of all the models in Table 6, wherein it is noticeable that our proposed technique results in a considerably lower error rate than the next best reported results. However, for the FVC2002-DB1 database, we obtained an identical EER = 0.19% to

Reference	Primary Component	FVC2002		FVC2004	
		DB1	DB3	DB1	DB3
Jin *et al*. [12]	Minutiae-based Bit-string	1.19	-	16.35	-
Wang and Hu [29]	DITOM	3.5	7.5	-	-
Jin *et al*. [11]	Randomized Graph-based Hamming Embedding	4.36	-	24.71	-
Wang and Hu [30]	Curtailed Circular Convolution	2	6.12	-	-
Wang *et al*. [28]	Partial Hadamard Transform	1	5.2	-	-
Wang *et al*. [31]	Zoned Minutia Pairs	0.19	4.29	-	-
Kho *et al*. [14]	Non-negative Least Square	2.28	6.4	-	-
Sadhya and Singh [23]	Cryptographic Hash Function	5.8	-	15.8	-
Sandhya *et al*. [24]	Delaunay Triangle	3.96	6.89	12.17	17.73
Jin *et al*. [9]	IoM Hashing (URP)	0.43	6.6	4.51	8.46
	IoM Hashing (GRP)	0.22	3.07	4.74	3.99
Baseline	**-**	**0.92**	**1.83**	**2.86**	**3.62**
Proposed	**LSC Features**	**0.19**	**1.44**	**1.28**	**2.72**

Table 6. Comparative analysis of our proposed model under the stolen-token scenario.

that published in [31]. The primary reason behind getting such satisfactory recognition rates is the use of MCC as the feature extractor and KPCA as the pre-processing scheme, both of which are state-of-the-art techniques. Hence, for an unbiased comparison, we have also included the results of the corresponding unprotected baseline models comprising of the binarized feature vectors. Interestingly, our proposed scheme decreases the system EER by a factor of 0.79, 0.21, 0.55, and 0.24 for the FVC2002-DB1, FVC2002-DB3, FVC2004-DB1, and FVC2004-DB3 databases respectively. This result is particularly important since there generally exists some trade-off between the security and performance aspects of biometric models [21, 18]. We were successful in improving upon the baseline results primarily due to the working principle of the LSH technique (stated in Section 3).

6. Security Analysis

Now we categorically analyze the various security aspects of our proposed cancelable model.

6.1. Non-invertibility

The property of non-invertibility states that it should be computationally hard for an adversary to invert the LSC features back to the binary fingerprint codes. In our case, we consider the binarized fingerprint vectors to be the base templates since our work is focused on exploring the applicability of the bit sampling based scheme for fingerprints. The adversarial success probabilities in performing such a task under the Attack via record multiplicity (ARM), Single hash attack (SHA), and Multi-hash attack (MHA) scenarios are stated in Table 7 [22]. For estimating an approximate measure of the worst-case probability, we choose the average parameter values of $b = 25$ bits, $k = 10$, $l = 200$, $T = 0.25$, $r = \frac{1}{T} = 4$, $n = 12$, and $x = 5$ bits. Among all the attack scenarios, the best adversarial success corresponds to SHA being employed under the stolen token scenario. The actual value can be computed as:

Scenario	Genuine Token	Stolen Token
SHA	$\left(\frac{1}{r \times 2^x}\right)^n \times \frac{(b-k)!}{b!}$	$\left(\frac{1}{r \times 2^x}\right)^n$
MHA	$\left(\frac{1}{r}\right)^{l \times n} \times \left[\frac{(b-k)!}{b!}\right]^l \times \prod_{i=1}^{n} \frac{1}{2^{x_i}}$	$\left(\frac{1}{r}\right)^{l \times n} \times \prod_{i=1}^{n} \frac{1}{2^{x_i}}$
ARM	$\left(\frac{1}{r}\right)^{l \times y \times n} \times \left[\frac{(b-k)!}{b!}\right]^{l \times y} \times \prod_{i=1}^{n} \frac{1}{2^{x_i}}$	$\left(\frac{1}{r}\right)^{l \times y \times n} \times \prod_{i=1}^{n} \frac{1}{2^{x_i}}$

Table 7. Adversarial success probabilities for non-invertibility analysis of the LSC features.

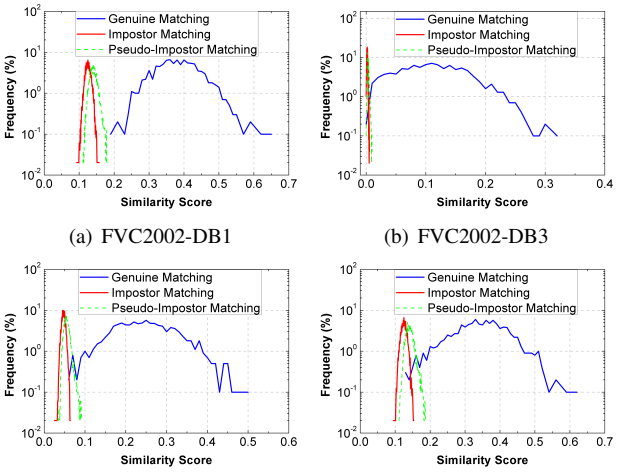
$$\mathbb{P} = \left(\frac{1}{r \times 2^x}\right)^n = \left(\frac{1}{2^2 \times 2^5}\right)^{12} = \frac{1}{2^{84}} \qquad (8)$$

Thus in the worst case, the adversary gains an advantage by a factor of ≈ 3.57 over the brute force approach. Although this quantity can be considered as computationally achievable under certain modern resources, we can increase this bound by extracting larger binary feature vectors from fingerprints. Moreover, the success probability stated in Eqn. (8) indicates only the non-invertibility of the LSC technique. An adversary would further require to invert the entire pre-processing scheme (including binarization and TKPCA) for obtaining the original minutiae point triplet values.

6.2. Revocability

Revocability dictates that multiple and distinct LSC features should be created from the same base fingerprint template. This guarantee can be empirically investigated by generating the corresponding *pseudo-impostor* distribution. In this scenario, 100 random permutation tokens (keys) were utilized for creating 100 distinct LSC features. The first LSC feature was subsequently matched with the remaining 99 features for generating the pseudo-impostor scores. The claim of revocability is validated if - (1) the genuine and pseudo-impostor distributions have a clear separation, and (2) the impostor and pseudo-impostor distributions overlap [30, 28]. These distributions for all the four databases are shown in Fig. 4, wherein both of the required characteristics can be distinctively observed.

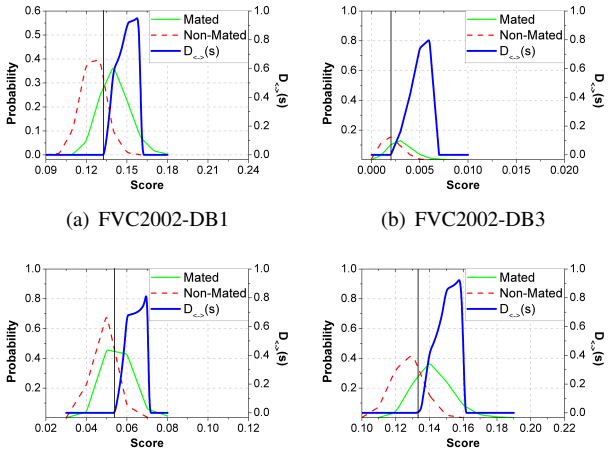
The separability (or overlap) between two distributions can be quantitatively estimated by the corresponding d', which is presented in Table 8 for all the three distributions. A higher value of d' indicates a good separation between the two distributions and vice-versa. These results hence suggest that the properties of the protected templates which are created from the same fingerprint sample are identical to that for the other classes. Therefore, we can empirically justify the fulfilment of revocability in our model.



(a) FVC2002-DB1 (b) FVC2002-DB3

(c) FVC2004-DB1 (d) FVC2004-DB3

Figure 4. Revocability analysis for the four fingerprint databases.



(a) FVC2002-DB1 (b) FVC2002-DB3

(c) FVC2004-DB1 (d) FVC2004-DB3

Figure 5. Unlinkability analysis for the four fingerprint databases. The solid black vertical lines represent $LR(s).\omega = 1$.

Database	Distribution		
	Genuine/ Impostor	Genuine/ Pseudo-Impostor	Impostor/ Pseudo-Impostor
FVC2002-DB1	5.65	5.28	1.65
FVC2002-DB3	2.73	2.7	0.99
FVC2004-DB1	3.68	3.53	1.36
FVC2004-DB3	3.92	3.6	1.73

Table 8. Decidability Index (d') of the genuine, impostor, and pseudo-impostor distributions.

6.3. Unlinkability

The unlinkability property states that different cancelable features which are generated from a single fingerprint sample should be indistinguishable from each other. This important property can be empirically examined by generating the *mated* (H_m) and *non-mated* (H_{nm}) scores. The mated scores are calculated by comparing the intra-class LSC templates which are generated by using different permutation tokens, whereas the non-mated scores correspond to LSC templates arising from different fingerprint sample using different keys. Based upon the likelihood ratio of these distributions, two novel measures have been defined for quantitatively measuring the degree of unlinkability: (i) Local measure $D_{\leftrightarrow}(s)$, and (ii) Global measure $D_{\leftrightarrow}^{sys}$ [7]. Both of these metrics have values in $[0, 1]$, with zero denoting complete unlinkability and one indicating no unlinkability. Following previous works, we created six cancelable databases from a fingerprint dataset by using distinct permutation tokens (while assuming $\omega = 1$). The resulting H_m, H_{nm}, and $D_{\leftrightarrow}(s)$ values are presented together in Fig. 5. The corresponding global measures were noted to be $D_{\leftrightarrow}^{sys} = \{0.016, 0.006, 0.015, 0.006\}$ for the FVC2002-DB1, FVC2002-DB3, FVC2004-DB1, and FVC2004-DB3 databases respectively. We can quantitatively justify the unlinkability property of our model since the $D_{\leftrightarrow}^{sys}$ values are very close to zero for all the four databases.

7. Conclusion

Fingerprint-based biometric models are one of the most commonly accepted means for verifying individual identities. Our study explores the feasibility of generating cancelable LSC templates from fingerprint samples by developing a working model. In our proposed framework, we have initially extracted MCC features from raw fingerprint images and subsequently binarized them by using the KPCA and zero-thresholding schemes. Finally, we have generated the protected LSC features by randomly sampling bits from these binary features. Since the proposed scheme is based on the principles of LSH, our method does not result in the degradation of the recognition accuracy. We have conducted extensive empirical tests on four benchmark fingerprint databases: FVC2002-DB1, FVC2002-DB3, FVC2004-DB1 and FVC2004-DB3. The corresponding results aptly justify the performance aspect of our model, for which we have obtained comprehensively lower error rates than other related state-of-the-art works. Furthermore, we have empirically justified other essential model requirements such as revocability and unlinkability. Hence our proposed framework simultaneously provides strong theoretical security guarantees and enhanced recognition rates.

References

[1] R. Cappelli, M. Ferrara, and D. Maltoni. Minutia cylinder-code: A new representation and matching technique for fingerprint recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12):2128–2141, Dec 2010.

[2] R. Cappelli, M. Ferrara, and D. Maltoni. Fingerprint indexing based on minutia cylinder-code. *IEEE Transactions*

on Pattern Analysis and Machine Intelligence, 33(5):1051–1057, May 2011.

[3] M. S. Charikar. Similarity estimation techniques from rounding algorithms. In *Proceedings of the Thiry-fourth Annual ACM Symposium on Theory of Computing*, STOC '02, pages 380–388, New York, NY, USA, 2002. ACM.

[4] K. H. Cheung, A. W.-K. Kong, J. You, and D. Zhang. An analysis on invertibility of cancelable biometrics based on biohashing. In *InternationalConference on Imaging Science, Systems, and Technology*, pages 40–45, 2005.

[5] F. Farooq, R. M. Bolle, T. Jea, and N. Ratha. Anonymous and revocable fingerprint recognition. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7, June 2007.

[6] M. Ferrara, D. Maltoni, and R. Cappelli. Noninvertible minutia cylinder-code representation. *IEEE Transactions on Information Forensics and Security*, 7(6):1727–1737, Dec 2012.

[7] M. Gomez-Barrero, J. Galbally, C. Rathgeb, and C. Busch. General framework to evaluate unlinkability in biometric template protection systems. *IEEE Transactions on Information Forensics and Security*, 13(6):1406–1420, June 2018.

[8] A. K. Jain, K. Nandakumar, and A. Nagar. Biometric template security. *EURASIP J. Adv. Signal Process*, 2008:113:1–113:17, Jan. 2008.

[9] Z. Jin, J. Y. Hwang, Y. Lai, S. Kim, and A. B. J. Teoh. Ranking-based locality sensitive hashing-enabled cancelable biometrics: Index-of-max hashing. *IEEE Transactions on Information Forensics and Security*, 13(2):393–407, Feb 2018.

[10] Z. Jin, M. Lim, A. B. J. Teoh, B. Goi, and Y. H. Tay. Generating fixed-length representation from minutiae using kernel methods for fingerprint authentication. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(10):1415–1428, Oct 2016.

[11] Z. Jin, M.-H. Lim, A. B. J. Teoh, and B.-M. Goi. A non-invertible randomized graph-based hamming embedding for generating cancelable fingerprint template. *Pattern Recognition Letters*, 42:137 – 147, 2014.

[12] Z. Jin, A. B. J. Teoh, T. S. Ong, and C. Tee. Fingerprint template protection with minutiae-based bit-string for security and privacy preserving. *Expert Systems with Applications*, 39(6):6157 – 6167, 2012.

[13] M. Kayaoglu, B. Topcu, and U. Uludag. Standard fingerprint databases: Manual minutiae labeling and matcher performance analyses. *CoRR*, abs/1305.1443, 2013.

[14] J. B. Kho, J. Kim, I.-J. Kim, and A. B. Teoh. Cancelable fingerprint template design with randomized non-negative least squares. *Pattern Recognition*, 91:245 – 260, 2019.

[15] C. Lee, J. Choi, K. Toh, S. Lee, and J. Kim. Alignment-free cancelable fingerprint templates based on local minutiae information. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(4):980–992, Aug 2007.

[16] C. Lee and J. Kim. Cancelable fingerprint templates using minutiae-based bit-strings. *Journal of Network and Computer Applications*, 33(3):236 – 246, 2010. Recent Advances and Future Directions in Biometrics Personal Identification.

[17] Y. Lee, Y. Chung, and K. Moon. Inverse operation and preimage attack on biohashing. In *2009 IEEE Workshop on Computational Intelligence in Biometrics: Theory, Algorithms, and Applications*, pages 92–97, March 2009.

[18] V. M. Patel, N. K. Ratha, and R. Chellappa. Cancelable biometrics: A review. *IEEE Signal Processing Magazine*, 32(5):54–65, Sep. 2015.

[19] F. Quan, S. Fei, C. Anni, and Z. Feifei. Cracking cancelable fingerprint template of ratha. In *2008 International Symposium on Computer Science and Computational Technology*, volume 2, pages 572–575, Dec 2008.

[20] N. K. Ratha, S. Chikkerur, J. H. Connell, and R. M. Bolle. Generating cancelable fingerprint templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4):561–572, April 2007.

[21] C. Rathgeb and A. Uhl. A survey on biometric cryptosystems and cancelable biometrics. *EURASIP Journal on Information Security*, 2011(1):3, Sep 2011.

[22] D. Sadhya and B. Raman. Generation of cancelable iris templates via randomized bit sampling. *IEEE Transactions on Information Forensics and Security*, pages 1–1, 2019.

[23] D. Sadhya and S. K. Singh. Design of a cancelable biometric template protection scheme for fingerprints based on cryptographic hash functions. *Multimedia Tools and Applications*, 77(12):15113–15137, Jun 2018.

[24] M. Sandhya, M. V. N. K. Prasad, and R. R. Chillarige. Generating cancellable fingerprint templates based on delaunay triangle feature set construction. *IET Biometrics*, 5(2):131–139, 2016.

[25] B. Scholkopf, A. Smola, and K. Muller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, July 1998.

[26] A. B. J. Teoh, D. N. C. Ling, and A. Goh. Biohashing: two factor authentication featuring fingerprint data and tokenised random number. *Pattern Recognition*, 37(11):2245 – 2255, 2004.

[27] S. Tulyakov, F. Farooq, P. Mansukhani, and V. Govindaraju. Symmetric hash functions for secure fingerprint biometric systems. *Pattern Recognition Letters*, 28(16):2427 – 2436, 2007.

[28] S. Wang, G. Deng, and J. Hu. A partial hadamard transform approach to the design of cancelable fingerprint templates containing binary biometric representations. *Pattern Recognition*, 61:447 – 458, 2017.

[29] S. Wang and J. Hu. Alignment-free cancelable fingerprint template design: A densely infinite-to-one mapping (ditom) approach. *Pattern Recognition*, 45(12):4129 – 4137, 2012.

[30] S. Wang and J. Hu. Design of alignment-free cancelable fingerprint templates via curtailed circular convolution. *Pattern Recognition*, 47(3):1321 – 1329, 2014.

[31] S. Wang, W. Yang, and J. Hu. Design of alignment-free cancelable fingerprint templates with zoned minutia pairs. *Pattern Recognition*, 66:295 – 301, 2017.

[32] W. Yang, J. Hu, S. Wang, and M. Stojmenovic. An alignment-free fingerprint bio-cryptosystem based on modified voronoi neighbor structures. *Pattern Recognition*, 47(3):1309 – 1320, 2014.