Cancellable Fingerprint Authentication Schemes for Bio-Crypto Applications

Wei Jing Wong

Faculty of Engineering, Computing, and Science
Swinburne University of Technology Sarawak Campus
Kuching, Malaysia

Submitted for the degree of Doctor of Philosophy

2016

To my family — who love and support.

Abstract

Conventional biometric systems store the biometric features in unprotected templates. Once these templates are compromised, the raw biometric data can be easily recovered and used maliciously, such as identity fraud. Biometric template protection provides a tangible solution to this critical issue by performing a specially designed transformation to shield the biometric template. Ideally, the protected template is generated in the way that it is mathematically impossible to derive the raw biometric data from the template. Besides, it is made revocable. Therefore, the adversary would not learn anything valuable even if the protected template is compromised. Two categories of biometric template protection include cancellable biometrics and biometric cryptosystems. One greatest challenge of designing a biometric template protection scheme is that the recognition performance after transformation should not deteriorate. It is also important to close the gap between theory and practice from various aspects, including performance, security and privacy, and computational complexity.

Among the many biometric identifiers, fingerprint is most widely used due to its high distinctiveness and collectability. In this thesis, a comprehensive framework of generating cancellable templates from fingerprints is proposed. In addition, a hybrid biometric template protection scheme is also demonstrated by applying the proposed cancellable fingerprint on existing biometric cryptosystem. The entire cancellable fingerprint generation scheme is divided into four phases.

With the minutiae already extracted from the fingerprint, the first phase is to convert the raw minutiae template defined by the International Organization for Standardization into minutia vectors through the multi-line code algorithm. Multi-line code is a minutia descriptor which describes a reference minutia based on the distribution of the neighbouring minutiae. Viewed from the three dimensional space (x-coordinate, y-coordinate and ridge orientation), the neighbouring minutiae are covered by multiple fixed-radius cylinders constructed on multiple straight lines to provide a thorough sweep on the vicinity. The multi-line code fingerprint template is invariant to translation and rotation and is, to certain extent, robust against scaling and non-linear local distortions.

After that, the variable-size and unordered multi-line code template is transformed into a fixed-length and ordered vector through minutiae set to feature vector (set-to-vector) transformation. Two distinct set-to-vector transformation methods, namely the kernel subspace analysis method and the bag-of-minutiae method, are introduced in this doctoral work. The former exploits the unique non-linear property of kernel principal components analysis and a specially designed kernel function that is adaptable to multi-line code template; the latter borrows the concept of bag-of-words modelling to perform vector quantization on the minutiae.

In order to fulfil the revocability attribute of biometric template protection, the fingerprint feature vector is then subjected to cancellable transformation. This phase allows the fingerprint template to be reissued by assigning a new user-specific key when the protected template is compromised. Although two cancellable transforms from biometric salting, viz. permutation and random projection, are available, random projection is found to be more superior than permutation in most aspects.

Finally, the fourth phase involves converting the real-valued template into bit-string. Two main streams of biometric template binarization are to be investigated, including static quantization and dynamic quantization. Static quantization assigns all vector components with the same number of bits, while its dynamic counterpart allocates different number of bits to each vector component depending on the discriminability of the component.

By applying the above described four-phase cancellable fingerprint generation framework with the fuzzy extractor biometric cryptosystem, a hybrid biometric template protection scheme, referred to as the cancellable fuzzy extractor, is constructed. It further enhances the security of biometric template protection while demonstrating the applicability of the proposed cancellable fingerprint.

Experimental results are presented based on four Fingerprint Verification Competition datasets to analyze the practicality of the proposed framework. Each of the phases discussed above are evaluated from the aspect of recognition accuracy, security and privacy, and computational complexity.

Acknowledgements

First and foremost, I give praises to God, the Almighty for providing me the opportunity, and granting me the wisdom and knowledge to proceed with this research project. The journey was long but I was not alone. I would like to express my thanks to all who have played a part in this project, and provided assistance and guidance throughout, both technically and fundamentally.

I would like to express my sincere gratitude to my supervisor, Associate Professor M. L. Dennis Wong, who has supported me throughout my Ph.D study. I appreciate his guidance and encouragement during my research and thesis writing whilst sharing his immense knowledge and skills in various areas generously. Besides, I would also like to acknowledge my external supervisor, Associate Professor Andrew Ben Jin Teoh. He has provided many valuable opinions and have always been giving constructive comments on my works. I admire his professional skills in related area and his selfless attitude towards sharing his ideas regarding this research project. I also want to thank my another supervisor, Dr. Yau Hee Kho, who has provided me much help when the project kick-started and has been patient in guiding me. This thesis would not have been completed without the superb supervisory team.

Further, I wish to thank Swinburne University of Technology (Sarawak Campus) for offering me the SUTS postgraduate research studentship (SPRS) which includes tuition fee waiver and monthly stipend. I would not have pursued my study without such tremendous financial support from the university.

The journey would be meaningless without my dearest family, who always been loving me so unconditionally, sharing my sufferings and joys all along. The greatest motivation that keeps me going comes from them. I pray for blessings upon them a hundredfold of what they have blessed me with.

Last but not least, I want to thank my friends and colleagues for upholding me alongside and creating a fun-filled and dynamic working atmosphere. These precious companions include Dr. Ming Ming Wong, Dr. Nguan Soon Chong, Kelvin Sheng Chek Yong, Lin Shen Liew, Zhi Hao Chang, Bih Fei Jong and the late Albin Sui Hian Kuek.

All glory and honour be to God.

Declaration

I hereby declare that, to the best of my knowledge, this thesis contains no material that has been submitted previously, in whole or in part, for the award of any other academic degree or diploma. Furthermore, any idea, technique, quotation, or any other material from other people's work included in this thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices.

_____

WEI JING WONG
2015

Contents

List of Figures

List of Tables

Commonly Used Acronyms

ATM	Automated teller machine
BCH	Bose, Chaudhuri and Hocquenghem
BoM	Bag-of-minutiae
BoW	Bag-of-words
BTP	Biometric template protection
CaFE	Cancellable fuzzy extractor
CIRF	Correlation-invariant random filtering
CPU	Central processing unit
DCT	Discrete cosine transform
DFT	Discrete Fourier transform
DNA	Deoxyribonucleic acid
DQ	Dynamic quantization
DWT	Discrete wavelet transform
EER	Equal-error rate
FAR	False acceptance rate
FFT	Fast Fourier transform
E*K*-SVD	Enhanced *K*-singular value decomposition
FRR	False rejection rate
FVC	Fingerprint verification competition
FE	Fuzzy Extractor
GF	Galois field
ICS	Intrinsic coordinate system
I*K*-SVD	Immune *K*-singular value decomposition
ISO	International standard for organization
K-SVD	*K*-singular value decomposition
KPCA	Kernel principal components analysis
LBP	Local binary pattern
LC-*K*SVD	Label-consistent *K*-singular value decomposition
LDA	Linear discriminant analysis
LoG	Laplacian of Gaussian

LRLED	Local relative location error descriptor
LTI	Linear time-invariant
MCC	Minutia cylinder code
MLC	Multi-line code
MLCN	Multi-line code generated using the number of neighbouring minutiae near the reference point
MLCD	Multi-line code generated using the mean of distances from the neighbouring minutiae to the reference point
MRC	Minutiae relation code
MsD	Multi-state discretization
MVD	Minutiae vicinity decomposition
NDTC	N-layer Delaunay triangulation net check
OMP	Orthogonal matching pursuit
PCA	Principle components analysis
PIN	Personal identification number
PSD	Positive semi-definite
RBF	Radial basis function
RFID	Radio-frequency identification
RGHE	Randomized graph-based Hamming embedding
RLE	Relative location error
ROI	Region of interest
RP	Random projection
S2V	Set to vector
SDK	Software development kit
SHA	Secure hash algorithm
SIFT	Scale-invariant feature transformation
SNR	Signal-to-noise ratio
SS	Secure sketch
STF	Square-tessellation-FingerCode
SVD	Singular value decomposition
SVM	Support vector machine
WFMT	Wavelet and Fourier-Mellin transform

CHAPTER 1

Introduction

1.1 Biometric Authentication System

Biometric authentication refers to recognizing or authenticating individuals using their personal characteristics, called the biometric traits or biometric identifiers. These characteristics are selected for authentication purpose based on their distinctiveness, universality, permanence and collectability. In general, biometric traits can be categorized into physiological traits and behavioural traits. Some commonly used physiological biometric traits are fingerprint, palm-print, face and iris; while behavioural traits include signature, voice, gait and keystroke. Additionally, soft biometrics [3] such as body weight, height, gender and eye colour can also be used to assist the authentication of individuals.

Generally, a biometric authentication system includes an enrolment process and an authentication process, as depicted in Fig. 1.1. The common steps involved in both of the processes are:

1. Pre-processing: The main purpose of the pre-processing step is to eliminate noises in the raw data. One or more image or signal processing techniques may be used, such as contrast stretching, filtering, segmentation and alignment.

2. Feature extraction: This step extracts useful information that represents the user's biometrics from the pre-processed data. These features vary depending on the biometric trait used, for example, singular points and minutiae are extracted from fingerprints, while freckles and crypts are possible features in irises.

3. Template generation: Template generation converts the biometric features extracted into vector or set of vectors representations, in either real, complex or bi-

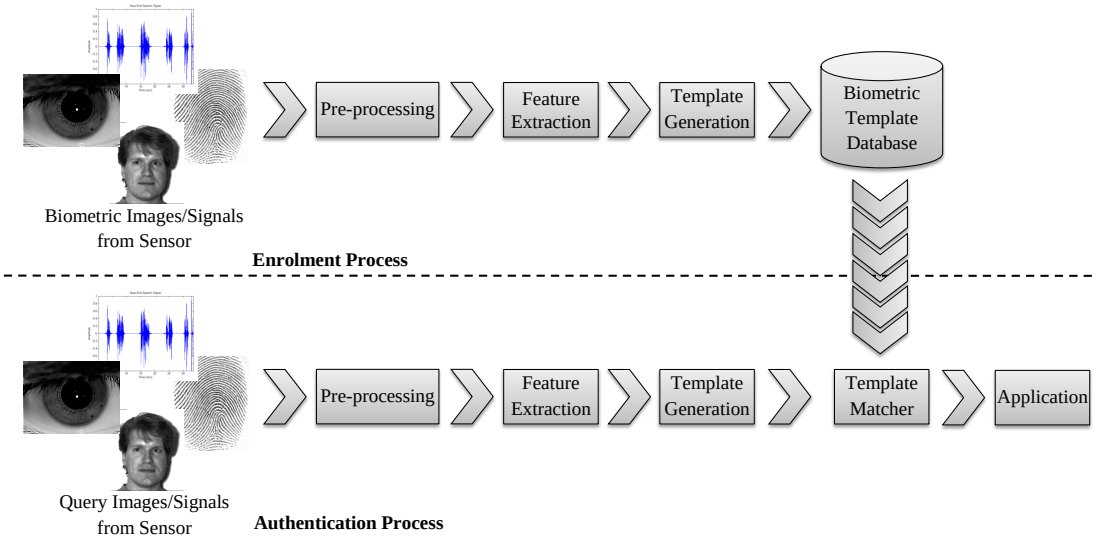


Figure 1.1: A general framework of a biometric authentication system.

nary form. Again, the method used depends on the biometric traits and features chosen. In some biometric recognition algorithms, such as correlation-based fingerprint recognition and principal components analysis- (PCA-) based face recognition, feature extraction and template generation are merged.

Finally, the templates generated in the enrolment process are stored in the biometric template database and are retrieved for matching upon authentication.

Biometric authentication offers numerous advantages over the conventional token-based and knowledge-based authentication such as RFID and PIN. First of all, it utilizes biometric traits which are part of the user's physical body or behaviour, thus is more convenient than conventional authentication cards and passwords which are easily lost, forgotten or stolen. Consequently, biometric authentication provides better security as it requires user to be present upon authentication. Unlike cards and passwords which may have duplicates, biometric traits are unique to individuals. Therefore, it can be used for not only verification but also person identification, especially in forensic and security access applications.

On the other hand, some security and privacy concerns have been raised against biometric authentication. One of the major threats in such systems is the compromise of the biometric template database. Conventional biometric template databases store the unprotected biometric features. These biometric features contain information about the raw biometric data and hence can be reversed to obtain the original images or signals of the biometric data effortlessly. Since biometric traits are permanent and irrevocable,

only one biometric template can be produced per trait per person. Once the biometric template is compromised, it can be used maliciously against the user, for instance, to access any biometric authentication system which employs the same biometric trait by presenting the compromised template to the matcher.

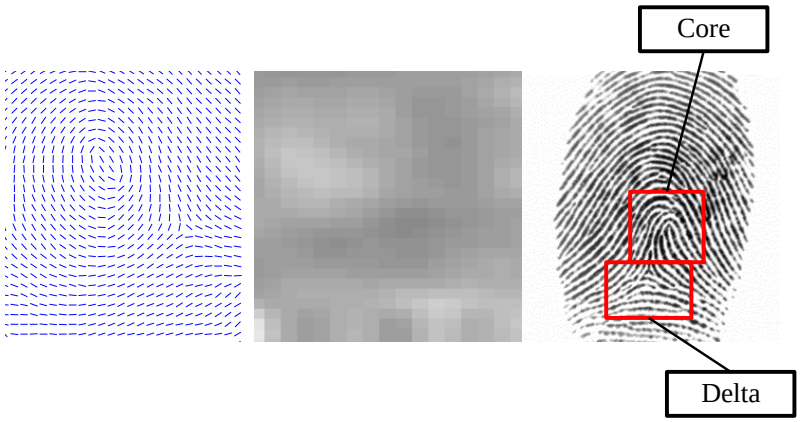
Furthermore, the recognition rate of biometric authentication depends on various factors, viz. the distinctiveness of the features extracted, the environmental and physical conditions upon authentication and the robustness of the template matching algorithms. For example, a genuine user may be rejected by a fingerprint recognition system (false rejection) due to new scars on his finger; or in another case, an impostor may be accepted by the system (false acceptance) if his features are similar to the genuine user's. Research works have been done to address other issues in biometric authentication, including detection of fake biometrics [4–6] and face recognition of monozygotic twins [7, 8].

An example of the application of biometric authentication is the Malaysian government multi-purpose card (also known as MyKad). MyKad is a national smart card which stores the thumbprint data of the owner on an embedded microchip and is designed with the following functions: identification, automated teller machine (ATM) transactions, passport information, health information and e-cash function [9]. However, these personal information, including the biometric data, could be easily accessible with a microchip scanner. Therefore, the greatest challenge of implementing MyKad is the privacy issues — in the context of this project, the privacy of the biometric data.

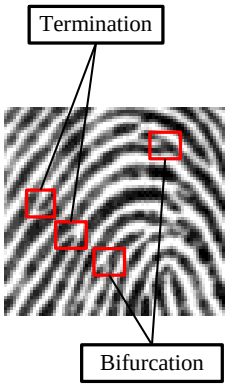
In a nutshell, according to the current stage of development, biometric authentication has yet to replace the conventional authentication methods as there exists drawbacks which hinder its usage in real-life applications. Research works to provide solutions to the issues are still on-going.

1.2 Fingerprint Recognition

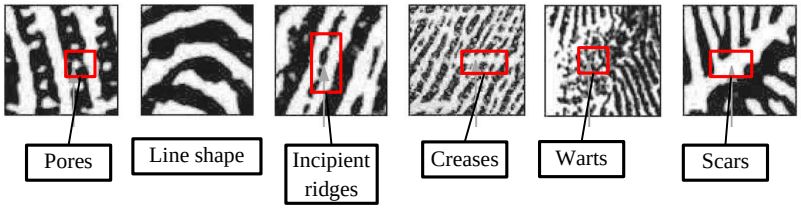
Fingerprints are the most commonly used biometric trait for authentication, in both security and forensic applications, due to its high distinctiveness and collectability. Fingerprint features are divided into three levels as shown in Figure 1.2. The level 1 fingerprint features consists of ridge orientation field and ridge frequency map in global sense. Subsequently, singular points such as cores and deltas and five different classes



(a) Level 1 features. The ridge orientation (first from the left) and ridge frequency (second from the left) are the fundamental features of a fingerprint. Higher gray-scale intensity in the ridge frequency map indicates higher ridge frequency. Singular points like *core* and *delta* (third from the left) can be obtained from these features.



(b) Level 2 features.



(c) Level 3 features. [11]

Figure 1.2: Three distinct levels of fingerprint features.

of ridge patterns [10] can be observed based on the level 1 features as depicted in Figure 1.3. The level 2 features are also called the minutiae, indicating the end points of ridges (terminations) and the points where ridges split (bifurcations). Minutiae are observed at the local level, thus are finer features compared to ridge orientations and frequencies. They are the most prevalent features used for fingerprint recognition. Viewing fingerprints at the finest level, the level 3 features capture even smaller details of the ridges, inclusive of sweat pores, shape of ridges, incipient ridges, creases, warts and scars. These features were proven to provide significant performance gain in fingerprint matching when combined with the previous levels of features [11]. However, level 3 fingerprint features can only be extracted from high resolution ($\geq$1,000 ppi) images and thus, are not applicable in standard fingerprint authentication systems, where the fingerprint resolution is 500 ppi.

Prior feature extraction, the fingerprint images, obtained in gray-scale, are usually put through some image enhancement processes to improve the quality of the fingerprint features, as illustrated in Figure 1.1. These pre-processing steps may include one or more of the followings:

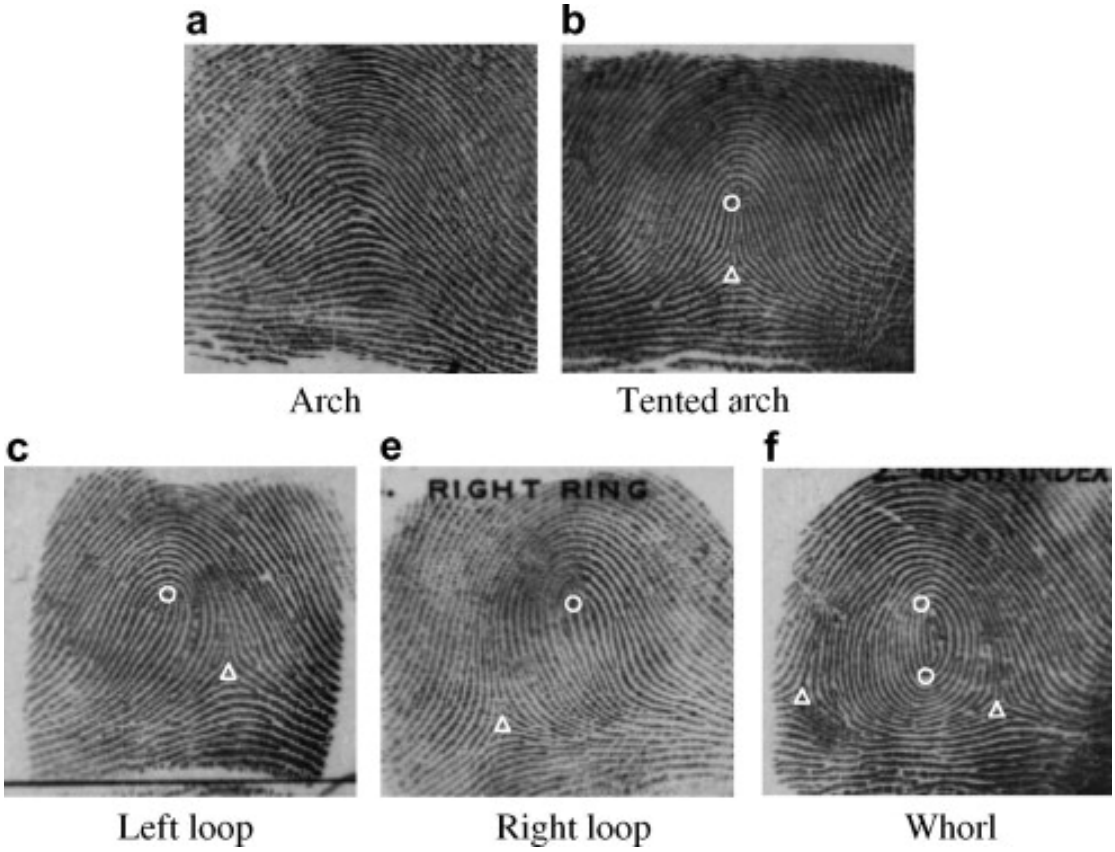


Figure 1.3: Five classes of fingerprint patterns derived from different positioning of the singular points, where ◯ indicates *core* and △ indicates *delta*. [1]

- Contrast stretching: to increase the gray-scale intensity variation between the darker pixels and the lighter pixels so that the ridges (darker) can be well-separated from the background (lighter). Contrast stretching techniques include histogram equalization, adaptive histogram equalization and image normalization.

- Ridge orientation field: to estimate the direction of the ridges as shown in Figure 1.2a. The ridge orientations also indicate the minutiae orientations, which is one of the important properties of the minutiae. Various methods have been proposed to estimate the orientation field of fingerprints, including the gradient-based method [12–16], model-based method [17, 18], spatial frequency content-based method [19] and estimation by ridge projection [20]

- Ridge frequency estimation: to measure the average distance between two ridges. The ridge frequency may be observed from the pixel intensity values across the line perpendicular to the ridge orientation [21].

- Filtering: to remove noises from the fingerprint image. Several filtering techniques have been used in the context of fingerprint image enhancement, includ-

ing Wiener filtering [22] and directional filtering such as Gabor filter [21, 23, 24], modified Gabor filter [25] and Log-Gabor filter [26] in either spatial, frequency or wavelet domain.

- Fingerprint segmentation: to locate the fingerprint region, or also known as the region of interest (ROI), in the image by evaluating the block-wise variance of the pixel values [14, 27]. A more extensive method for fingerprint region estimation was proposed by Hong et al. [21] by classifying image blocks based on ridge amplitude, ridge frequency and pixel variance within the blocks. Besides, Chikkerur et al. [28] employed thresholding on energy map for fingerprint segmentation.

Fingerprint matching algorithms are categorized into two major approaches, namely minutiae-based matching and correlation-based matching. Minutiae-based matching, as the name suggests, utilizes mainly the minutiae (level 2 features) in fingerprint matching. After the pre-processing steps, minutiae can be extracted from the enhanced fingerprint image through direct gray-scale methods [29, 30] or skeleton image-based methods [14, 31, 32]. The latter requires image binarization and thinning prior minutiae detection. The minutiae are represented in the three-dimensional (x-coordinate, y-coordinate and orientation) format (also known as the ISO template) and can be used for direct fingerprint matching or further template generation. On the other hand, correlation-based matching exploits the correlation of each image pixel in fingerprint matching. Algorithms proposed by the researchers for both fingerprint matching approaches are further discussed in section 2.1.

1.3 Biometric Template Protection

Biometric template protection (BTP) is one of the promising solutions to mitigate the aftermaths due to the compromise of biometric template database mentioned in section 1.1. For example, Cappelli et al. [33] have shown that it is possible to reconstruct fingerprint image from the ISO template. The idea of BTP is to transform the conventional unprotected biometric template into a protected template so that the adversary is not able to retrieve any useful information about the original biometric data from the protected template. In general, BTP schemes are designed to fulfil the following objectives [34]:

- Irreversibility: it should be computationally difficult to reconstruct the original biometric template from the protected template. Irreversibility and non-invertibility are interchangeable in this thesis.
- Revocability/diversity: different protected templates can be generated based on the same biometric features for different applications. One version of protected template should not match any of the other versions.
- Performance: the scheme should not deteriorate the recognition accuracy.

BTP schemes are categorized into cancellable biometrics and biometric cryptosystems. Cancellable biometrics apply systematic distortions to the biometric features to obtain the protected template so that a new template can be reissued by altering certain parameters in the transformation. In this case, biometric templates are compared in the transformed domain. On the contrary, biometric cryptosystems are helper data-based schemes which either generate a secret key from the biometric template (key-generation) or bind a secret key to the biometric template (key-binding). In biometric cryptosystems, a public helper data is stored without compromising the secret key and the biometric template. The secret key is released only if the query biometric data is close enough to the enrolled biometric data. Specific BTP schemes are discussed in section 2.2.1 to section 2.2.3.

1.4 Research Problems, Objectives and Contributions

As fingerprint biometrics is becoming more common in security applications such as national IDs, immigration systems, door access systems and secret data access systems, the possible security and privacy threats have drawn much concern from the researchers. BTP is an emerging concept of securing biometric data which is also one of the promising countermeasures against security and privacy breaches when the biometric template database is compromised. As cancellable biometrics [35] and biometric cryptosystems [36] were first introduced less than a couple of decades ago, there still exists challenges in the deployment of BTP schemes in real-life biometric authentication systems.

According to an on-line evaluation system for fingerprint recognition algorithms known as FVC-onGoing [37], the recognition performance of secured fingerprint templates has not caught up with that of unprotected templates. Hence, FVC-onGoing has included

secure template fingerprint verification as one of the benchmarks for accuracy and efficiency evaluation and comparison to encourage studies in this area. From the aspect of system security and user's privacy, existing BTP techniques are, to certain extent, vulnerable to one or more of the known attacks, viz. zero-effort attack, hill climbing attack, masquerade attack, reverse attack and brute-force attack. The aforementioned challenges deal with the main objectives of designing BTP schemes as discussed in section 1.3. Furthermore, BTP techniques often introduce increased computational complexity to template generation process and thus, require longer run-time during fingerprint enrolment and authentication.

This thesis proposes a hybrid BTP scheme by combining both cancellable biometrics (or specifically cancellable fingerprint) and biometric cryptosystems to provide a security-, privacy- and performance-oriented solution for biometric template database attacks. The fact that biometric cryptosystems do not require the storage of the biometric template is an advantage to the users' privacy of a biometric system. The main goal of this study is to create a cancellable fingerprint generation scheme that can be integrated into biometric cryptosystems. Except the requirements of BTP schemes listed in section 1.3, another criterion of applying a fingerprint template in some biometric cryptosystems, such as secure sketch and fuzzy extractor, is that the template must be a bit-string.

The major contributions of this thesis are declared as follows:

1. A minutiae-based non-invertible transform known as the multi-line code (MLC) is presented. MLC is a minutia descriptor which transforms the original ISO representation of minutiae into a high-dimensional minutia vector in such a way that the transformation is irreversible. The matching of MLC-based fingerprint templates (hereafter referred as MLC templates) does not require fingerprint pre-alignment, hence reduces the computational complexity during authentication. An efficient algorithm for MLC template generation is also presented to speed up the process.

2. The proposed MLC template is an unordered set of minutia vectors and its size depends on the minutiae extracted from the input fingerprint image. In order to adapt to biometric cryptosystems such as fuzzy commitment [36], secure sketch and fuzzy extractor [2], the biometric template needs to be in the form of a global

fixed-length vector [1]. In this thesis, we propose two novel minutiae set to feature vector (S2V) conversion methods to convert the MLC template into an ordered and global fixed-length vector including the kernel PCA- (KPCA-) based method and the bag-of-minutiae (BoM) modelling method. In the KPCA-based method, a new kernel is derived to suit the input template; in BoM modelling, the concept of bag-of-words (BoW) modelling is slightly modified to describe minutiae sets from the aspect of vector quantization.

3. Two cancellable transformations are used to realize the revocability of MLC template, namely permutation and random projection (RP).

4. Two alternatives of biometric template binarization, namely static quantization and dynamic quantization (DQ), are adopted to generate bit-string from the proposed fingerprint template.

5. A hybrid BTP scheme is demonstrated by combining the proposed cancellable fingerprint with fuzzy extractor.

1.5 Thesis Organization

This section provides an overview of the rest of the chapters in this thesis. Chapter 2 reports the existing works on both fingerprint matching and BTP schemes in categories. In Chapter 3, the MLC template generation algorithm is presented, alongside with two cancellable transformations, to create a cancellable fingerprint. After that, the two aforementioned S2V transformation methods are presented in Chapter 4 (KPCA-based method) and Chapter 5 (BoM modelling) respectively. The cancellable fixed-length fingerprint template is then converted into bit-string through static quantization and DQ in Chapter 6. The amalgamation between cancellable fingerprint and biometric cryptosystems is established in Chapter 7. In each of these chapers, a table explaining the nomenclature of important symbols used is provided for the convenience of referencing. To complete the thesis, Chapter 8 summarizes the contents and results obtained in the previous chapters and proposes some possible future works for this study.

[1] Note that some methods in the literature, such as minutia cylinder code (MCC) [38], produce *local* fixed-length vector, that is, all minutia vectors possess the same length, but the size of the fingerprint template varies in *global* sense.

CHAPTER 2

Literature Review

2.1 Fingerprint Matching

Figure 2.1 shows the various categories and subcategories of fingerprint matching algorithms. Existing methods are discussed according to the corresponding category or subcategory in this section.

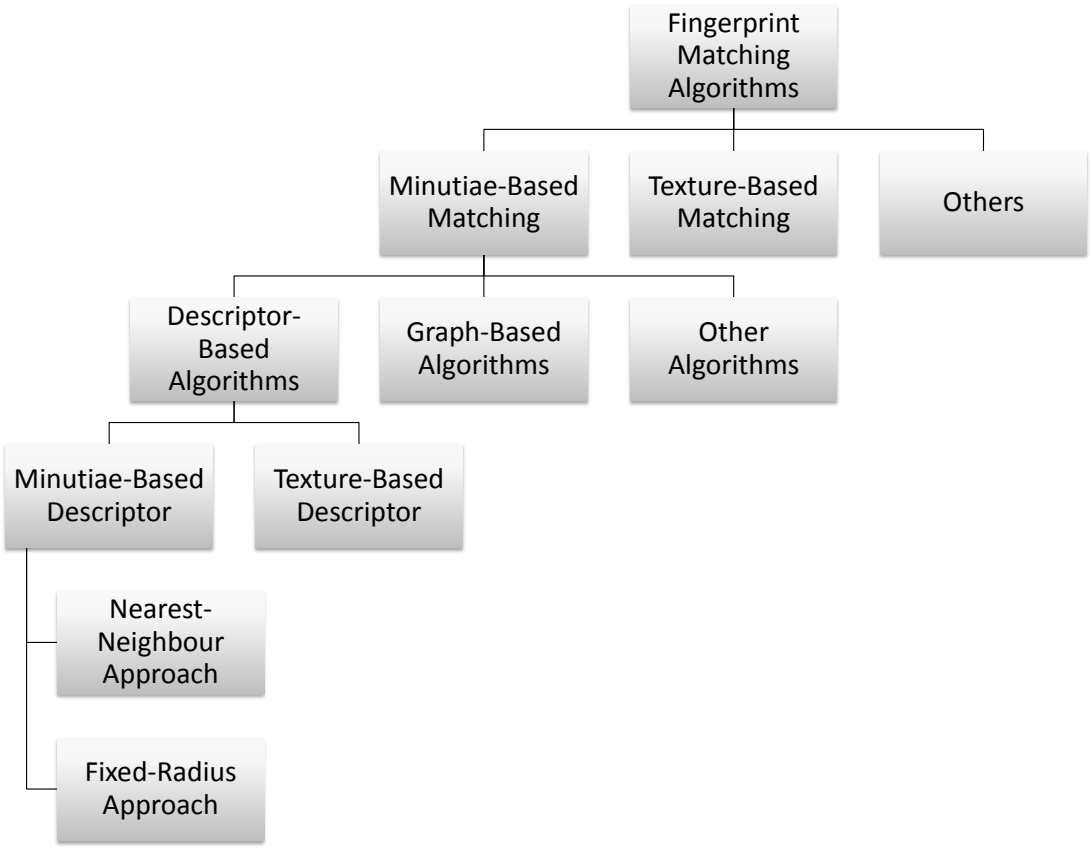


Figure 2.1: Hierarchy of categorization of fingerprint matching algorithms.

2.1.1 Minutiae-Based Fingerprint Matching

Descriptor-Based Algorithms

One common approach toward minutiae-based fingerprint matching is by minutia descriptor. Minutia descriptor refers to the methodology of extracting a multi-dimensional vector by capturing distinguishing information that is unique to individual minutia. The resulting vector is coined as the minutia vector. Multiple minutia vectors can be produced from a fingerprint and are used for matching. Minutia descriptors are subdivided into minutiae-based descriptor and texture-based descriptor. The former exploits the neighbouring minutiae while the latter utilizes the surrounding ridge patterns to describe the reference minutia.

Jiang and Yau [39] presented a nearest-neighbour approach to describe a minutia using the relative distance, radial angle, relative orientation, ridge counts and minutia types of the neighbouring minutiae. Lee et al. [40] adopted the same technique but included an additional feature — ridge frequency of the neighbouring minutiae. Similarly, Jea and Govindaraju [41] used the relative distance, relative orientation and the angle between the segments connecting the two nearest neighbours and the reference minutia as the features. In these methods, minutia descriptor is used for fingerprint alignment and the aligned minutiae are then matched globally in the three-dimensional (x-coordinate, y-coordinate, orientation) domain.

Chikkerur et al. [42] defines a novel nearest neighbour-based descriptor that utilizes k minutiae in the neighbourhood as a k-plet. The measurements included in a k-plet are identical to those used in other nearest neighbour-based descriptors [39–41], however the minutiae are selected so that a nearest neighbour is chosen in each of the four quadrants sequentially. Moreover, no fingerprint alignment is required in this technique; dynamic programming and newly proposed coupled breadth first search are used for fingerprint matching and local matches consolidation.

Furthermore, the local relative location error descriptor (LRLED) was proposed by Tong et al. [43] to match fingerprints using the relative location error (RLE) of the minutiae. In this method, the ISO template of fingerprints are stored and used for LRLED evaluation upon authentication. After searching for the potential corresponding minutiae set based on merely the distance between pre-aligned minutiae, the RLE of the corresponding minutiae pairs are computed and matched. This method requires ad-

ditional computational power in the authentication process, where pre-alignment and descriptor generation are executed.

The minutia descriptors discussed above are nearest-neighbour minutiae-based descriptors. Such technique is sensitive to missing and spurious minutiae as a missing or spurious minutia may result in significant difference in the minutia vector extracted. An alternative to the nearest-neighbour approach is the fixed-radius approach.

Hrechak and McHugh [44] proposed a feature descriptor which captures the occurrence of different types of features within a fixed radius from the central feature. Besides the minutiae (ridge terminations and bifurcations), other features including dots, islands, spurs, crossovers, bridges and short ridges are also taken into account. These features are difficult to identify as some are actually formed due to distortions in the fingerprint image and thus, are not reliable. Wahab et al. [45] proposed a similar method, of which five minutiae within a predefined radius that are nearest to the central minutia are selected. The attributes stored include the minutia type, x-coordinate and y-coordinate of the central minutia and the minutia type, distance, relative angle and the ridge count of the neighbouring minutiae. Correlation matching [46] is used to match two fingerprints, both locally and globally. Another fixed-radius minutiae-based descriptor named minutia vicinity was presented by Bringer and Despiegel [47]. In each minutia vicinity, all minutiae within the radius are aligned according to the central minutia. The neighbouring minutiae are matched to obtain the micro-scores. The local score between two vicinities and subsequently the global score between two fingerprints are obtained by applying the Hungarian algorithm [48] twice.

A state-of-the-art minutiae-based descriptor coined as the minutia cylinder-code (MCC) was proposed by Cappelli et al. [38]. A tessellated cylinder is constructed centring at each minutia on the three-dimensional space. In order to address the close-to-border-minutiae problem of fixed-radius method, a radial basis function (RBF) is used to calculate the contribution of neighbouring minutiae to the cells. MCC is, by far, the best-performing minutiae-based representation of fingerprint with equal-error rate (EER) of 0.15% on FVC2006 datasets.

All the minutia descriptors mentioned above are minutiae-based descriptors, in which a minutia is described by using the information of other minutiae. Tico and Kuosmanen [49] introduced a texture-based descriptor based on circular pattern around the reference minutia. Sampling points are marked on circles of different radius centred

at the reference minutia and the orientation difference between the reference minutia and the sampling points are recorded to form the minutia vector. A similar method was presented by Qi and Wang [50], of which multiple lines of different direction are constructed on the reference minutia to sample the neighbourhood orientation instead of circles. The same concept was adopted by Wang et al. [51] to form OrientationCode and additionally, PolyLines, which observe the curvature of the ridges connecting to the minutiae. Moreover, another texture-based minutia descriptor which captures the information of the ridges attached to the neighbouring minutiae was proposed [52].

On top of the single-type descriptor-based algorithms, Feng [53] introduced the hybrid descriptor combining texture-based descriptor [49] and fixed-radius minutiae-based descriptor. Besides, 17 distinct features are proposed to offer more accurate matching score calculation and support vector machine (SVM) is applied for fingerprint classification. Furthermore, a scale-invariant feature transformation- (SIFT-) based minutia descriptor was proposed by Zhou et al. [54,55]. A histogram-based descriptor was proposed by Aggarwal et al. [56] which captures the gradient histogram of the sub-regions in every minutia block.

Graph-Based Algorithms

A novel minutiae graphing-based fingerprint matching technique based on Delaunay triangulation was first introduced by Bebis et al. [57]. Invariants such as the length of three edges and the cosine of the three angles of each triangle are used for matching. In addition to that, Parziale and Niel [58] and Liu et al. [59] used the orientation of the vertices and the direction of the edges to perform alignment and fingerprint matching. Yang et al. [60] took an extra step to eliminate unmatched triangles and provide robust minutiae matching by introducing the N-layer Delaunay triangulation net check (NDTC) algorithm. Besides, Xu et al. [61] performed fingerprint matching by fusing Delaunay triangles.

Another graph-based method is by using Voronoi diagram [62]. Fingerprint alignment is done by matching the central cell and the aligned minutiae are matched globally in the three-dimensional space. A fingerprint matching algorithm incorporating both Delaunay triangulation and Voronoi diagram has also been presented [63] and has established lower EER than both individual graph-based methods.

The advantage of using graph-based fingerprint matching is that the attributes of the shapes (triangles or polygons) generated are rotation- and translation-invariant. However just like nearest-neighbour minutia descriptors, minutiae graphing is implemented based on the positional correlation among minutiae and thus, is susceptible to missing and spurious minutiae.

Other Minutiae-Based Algorithms

Another widely researched area in minutiae-based fingerprint matching is minutiae matching with pre-alignment. Fingerprint pre-alignment techniques include core point-based pre-alignment [64, 65], orientation field-based pre-alignment [66, 67] and ridge-based pre-alignment [14, 68]. Hybrid systems such as pre-alignment using orientation field, ridge frequency and ridge curvature [69] and combination of phase-only correlation and core point-based pre-alignment [70] have also been proposed. However in general, fingerprint pre-alignment is time-consuming and is not able to address local distortions. As such, the performance of fingerprint matching through pre-alignment cannot match with that of the descriptor-based methods.

In order to avoid pre-alignment, Bazen and Gerez [71] introduced a new intrinsic coordinate system (ICS) that partitions the fingerprint into four regions based upon the ridge patterns. The minutiae positions are redefined and matched according to the ICS.

2.1.2 Texture-Based Fingerprint Matching

One of the pioneering texture-based fingerprint matching method is the FingerCode [23, 72]. Eight directional Gabor filters are applied on the radially tessellated sectors centred at the reference point to obtain the FingerCode. The reference point is defined as the point of maximum curvature of the concave ridges in the fingerprint image, hence the method is translation-invariant. Another advantage of this method is that it is able to produce fixed-length fingerprint template, which is desirable in biometric cryptosystems. Sha et al. [73] improved the method by considering the orientation information to provide a rotation-invariant FingerCode. Other variants of FingerCode include the square-tessellation-FingerCode (STF) [27], the interpolation-based STF [74] and the minutiae-based FingerCode [75].

Teoh et al. [76] proposed a fingerprint matching technique based on wavelet and Fourier-Mellin transform (WFMT) to address common errors in fingerprint feature detection

caused by translation, rotation and scaling. The low frequency sub-band of the fingerprint image is first obtained using wavelet decomposition and subjected to fast Fourier transform (FFT) and log-polar transform to generate the WFMT features. Other similar techniques based on time-frequency analysis are the multi-resolution discrete wavelet transform (DWT) [77,78] and discrete cosine transform (DCT) [79]. On the other hand, local binary pattern (LBP) features [80] and Hu's invariant moments [81,82] has also been used for texture-based fingerprint matching.

2.1.3 Other Fingerprint Matching Algorithms

A ridge-based matching was proposed by Xie et al. [83] by extracting ridges from the skeleton image and comparing neighbouring ridge information. Short ridges and closed ridges caused by pixel glitches in the fingerprint image may cause deterioration in the performance of this ridge matching method. Therefore, Feng et al. [84] employed a post-processing step to regularize the ridge pattern to eliminate ill-formed ridges.

Instead of using the minutiae, Park et al. [85] used SIFT points and the gradient information [56] around the points to perform fingerprint matching. The results showed that although using SIFT points alone does not improve the EER, the fusion between SIFT points matcher and minutiae matcher yield better accuracy than each individual matcher.

The combination of singular points (cores and deltas) and orientation field has also been used to provide translation- and rotation-invariant fingerprint matching [86–88]. However, such method is not suitable in cases where no singular point is detected due to poor fingerprint image quality, partial fingerprint or unique fingerprint pattern (*arch* class as shown in Figure 1.3). Besides, level 3 fingerprint features [11,89–91] has been incorporated in fingerprint matching.

2.1.4 Summary

In general, minutiae-based matching is more robust than texture-based matching against local non-linear distortions as it considers local structural details without neglecting global uniformity and continuity. Besides, minutiae carry most of the fingerprint discriminatory information [34]. Among the minutiae-based algorithms discussed, fixed-radius minutia descriptor-based matching is able to handle missing and spurious minutiae better than the others. Also, most of the descriptor-based methods introduce

translation- and rotation-invariant features and thus, eliminate the hassle of fingerprint pre-alignment.

However in the case of partial fingerprints, texture-based and other non-minutiae feature-based methods are more superior than minutiae-based methods as the number of extractable minutiae may be extremely small and insufficient for the generation of definitive feature vector. In addition, poor quality fingerprints such as latent fingerprints may affect the reliability of the minutiae extracted and subsequently, the accuracy of minutiae-based fingerprint matching.

2.2 Fingerprint Biometric Template Protection (BTP)

2.2.1 Cancellable Biometrics

Since cancellable biometrics stresses on two imperative attributes namely non-invertibility and revocability, the categorization of cancellable biometric schemes depends on the these enabling mechanism behind the attributes. Two main categories of cancellable biometrics include non-invertible transforms and biometric salting [92]. The former category applies non-invertible transform function to the biometric data so that the original biometric data cannot be reconstructed even if the cancellable template and transform method are compromised. The revocability of this approach is realized by modifying the parameters of the transform function. On the other hand, biometric salting applies transform which may be invertible [93] if the user-specific secret key is compromised. In this case, the biometric template subjected to the transform may be extracted through a non-invertible method. The secret key assures the uniqueness of the transformed template between users and has to be presented upon authentication. In this subsection, the instances of cancellable biometric schemes are discussed in the context of fingerprint biometrics according to each belonging category.

Non-invertible Transforms

Most of the non-invertible transforms involve spatial perturbation of the minutiae. Such perturbation can be further divided into projection-based perturbation, function-based perturbation and block remapping. Projection-based perturbation projects the minutiae onto a predefined two-dimensional plane with randomized position and direction; function-based perturbation derives a mathematical function with randomized

parameters to alter the original minutiae space; lastly, block remapping divides the minutiae space into multiple blocks and scrambles the positions of the blocks. All the perturbation methods are designed in the way that they are many-to-one mapping.

Ang et al. [94] presented a geometric transform based on the reflection of minutiae. In this approach, a line passing through the core point is drawn, and the minutiae below the line are reflected while the minutiae above remain. The gradient of the line is determined by a user-specific key ranges from 0 to π. Confusion occurs when dealing with fingerprints with no core point (i.e. arch) and with more than one core points (i.e. whorl). Also, since only one side of the minutiae are reflected, the final template still retains a part of the original minutiae set and thus weakens the security. Other geometric projection-based methods [95, 96] which project the minutiae onto a circle have also been proposed.

On function-based perturbation, Tulyakov et al. [97] used symmetric hash functions to convert the minutiae into hash values. In this algorithm, a minutia is represented by a complex number c_i. For each minutia in the fingerprint, a triplet (c_i, c_j, c_k) is formed with its two nearest neighbouring minutiae and is hashed using predefined hash functions. A secret key is introduced to seed the choices and order of hash functions for different fingerprints. This work was extended by combining more than one hash functions during implementation to increase the security of the template [98]. Also, k-plets of minutiae were used instead of triplets, where k can be more than three. Although it is impossible to reverse the hashed data, a large number of high power hash functions are needed to ensure the revocability of the template, which leads to high complexity.

Lee et al. [99] proposed an alignment-free cancellable fingerprint generation method by extracting invariant features following the same fashion used by Tico and Kuosmanen [49]. Together with a user-specific PIN, the invariant features are used to parametrize two changing functions which contribute to the perturbation of the minutiae, namely the distance-changing function and the orientation-changing function. Another cancellable template utilizing invariant features based on triplets was proposed by Farooq et al. [100]. The features measured are the length of the three sides, the orientations of the three vertex minutiae and the height of the longest side of a triplet. The template is a binary string of quantized feature values, so it requires less database storage.

Ratha et al. [101] proposed three kinds of non-invertible transforms including Cartesian transform, polar transform and functional transform. Cartesian transform and

polar transform divide the fingerprint space into cells of equal size and rearrange the minutiae according to the cell they belong to on a many-to-one mapping basis. Functional transform applies a spatial distortion to the fingerprint space using Gaussian kernels so that the position of the minutiae are translated and rotated in the same way. The first two transforms are instances of block remapping while the third method is a function-based perturbation. However, researchers pointed out that these transforms are vulnerable to attacks as most of the transformed minutiae are possible to be reversed to their original locations [102].

As one of the best-performing fingerprint representations, MCC [38] has been elevated to protected MCC (p-MCC) [103] as a template protection scheme to enhance the security of MCC. However, p-MCC is not revocable. Zhang et al. [104] proposed a cancellable fingerprint template generated based on MCC by sectioning and remapping the original MCC, assisted by a random MCC-like structure called the random plate. This method inherits the accuracy of MCC and is able to achieve <0.1% EER even in stolen-key scenario.

Besides, Yang et al. [105] performed polar-based block remapping on Delaunay triangles. Since the Delaunay triangles are represented by the invariants [58], this method provides better security than direct polar-based transform on the minutiae [101]. Wang and Hu [106] also developed an alignment-free cancellable fingerprint using minutiae pairs and curtailed circular convolution.

Other than the spatial perturbation approach, a unique histogram-based method was presented by Sutcu et al. [107]. It is also called a local point aggregation approach which constructs random cuboids on a three-dimensional space (x-coordinate, y-coordinate and orientation) and count the numbers of minutiae within the cuboids. The length of the feature vector is determined by the number of cuboids generated. The final bit-string is obtained through median-based thresholding. The template is revocable as different sets of random cuboids can be used to generate multiple distinct templates from one fingerprint. One notable advantage of this method over aforementioned methods is that it produces fixed-length representation for fingerprints. However, fingerprint pre-alignment is required prior template generation. This work was extended by Nagar et al. [108], in which more discriminative features within the cuboids were used, such as the distance from minutiae to the nearest boundary, the average and standard deviation of minutiae coordinates.

Furthermore, a combination of projection-based spatial perturbation and histogram-based method was introduced by Ahmad et al. [109]. In this method, the minutiae are projected onto a line crossing the core point with its slope determined by a user-specific key. The projected minutiae are partitioned into groups to generate a histogram-like vector.

Biometric Salting

Two major technique used for biometric salting are random projection (RP) and permutation. As mentioned before, while these two techniques may be invertible, the fingerprint features can be extractable in the way that it is infeasible to reconstruct the original biometric data.

BioHashing (or particularly known as FingerHashing for fingerprint biometrics) [110, 111] is one of the pioneers in RP-based biometric salting. It is a two-factor transform that employs the WFMT features of fingerprints [76] and a user-specific tokenized key to seed the random matrix for projection. User-dependent multi-state discretization [112] was used in the generation of binary bit-string to improve the performance of FingerHashing specifically for stolen-token scenario. Since WFMT produces a fixed-length feature vector, FingerHashing, in the same nature, produces a fixed-length vector while being a cancellable fingerprint generation scheme.

Another RP-based method was presented by Wang and Hu [113] known as the densely infinite-to-one mapping approach. In this method, the invariant features of every minutiae pair are quantized and a histogram is generated. The histogram is then converted into a complex vector by using discrete Fourier transform (DFT). The DFT features are subjected to RP for revocability. This approach excels in security, even when the template and the parametric key are stolen.

Jin et al. [114] presented a graph-based cancellable fingerprint, dubbed as the randomized graph-based Hamming embedding (RGHE). RGHE embeds a set of randomized invariant features generated by minutiae vicinity decomposition (MVD) [115] and RP into a Hamming space to obtain a bit-string for each minutiae vicinity. Although Hamming embedding has been found to preserve the accuracy of MVD, RGHE yields much higher EER for lower fingerprint image quality.

On the permutation side, a novel bit-string representation of fingerprint template was introduced by Lee and Kim [116], in which each minutia is described by a three dimen-

sional array. The width and height of the three dimensional array is the x-y plane of the fingerprint image, whereas the depth represents the orientation of minutiae. The array is divided into cells of equal size, and the number of minutiae in the cells form the final bit-string via zero-thresholding. It provides high revocability by using simple permutation, which might be easily reversed if the permutation order is exposed. A similar approach was presented where a polar grid instead of a cuboid was used to quantized the neighbouring minutiae [117].

Moreover, Jin et al. [118] utilizes the invariant features in minutiae pair representation to generate fixed-length vector. These invariant features are then quantized and indexed to obtain a single integer per minutiae pair. The final vector is a permuted histogram generated based on the indexed minutiae pairs.

Yang et al. [119] has suggested dynamic random projection to enhance the security of the conventional RP used by FingerHashing. The idea is to construct a random matrix dynamically, depending on the biometric feature vector itself.

In addition, a unique fingerprint salting approach was proposed by [120]. The transformation utilizes chip matching algorithm [121] based on correlation-invariant random filtering (CIRF). It first extracts chip images centred at the minutiae from the fingerprint image and transform these chip images using CIRF to generate the template. The method stresses on the security and privacy of cancellable fingerprint template. The mathematical properties of CIRF were further investigated by [122] to derive a new algorithm for cancellable biometrics that establishes better security without affecting the accuracy.

By combining the concept of FingerHashing and FingerCode, Belgeuchi et al. [123] presented a minutiae-based fingerprint salting scheme. The proposed method extracts the FingerCode for every minutia and applies FingerHashing on the FingerCode to produce a protected minutia template, coined as BioCode. Furthermore, minutiae k-plets are used as additional information in template generation [124,125]. Another cancellable fingerprint scheme utilizing the FingerCode, referred as BioPhasor [126] was also introduced. In this method, the FingerCode is binarized according to quantization level determined by a tokenized pseudo-random number.

2.2.2 Biometric Cryptosystems

Biometric cryptosystems are classified into biometric key-generation and biometric key-binding schemes. The former aims at deriving a cryptographic key and helper data from a given biometric template; as opposed to that, biometric key-binding schemes use a biometric template to secure a chosen key and helper data is derived from both the key and the biometric template. In this subsection, examples of biometric cryptosystems, specifically for fingerprint biometrics, are discussed.

Key-Generation Biometric Cryptosystems

Dodis et al. [2] introduced two biometric cryptographic key-generation models, namely secure sketch (SS) and fuzzy extractor (FE). A FE can also be defined as a combination of a SS and a strong randomness extractor as depicted in Figure 2.2. It is able to address both error tolerance and non-uniformity in the biometric data. Three main techniques were proposed for building a SS scheme, viz. constructions for Hamming distance, set difference and edit distance. Among the three SS builds, constructions for set difference, for example the notable PinSketch which is constructed based on syndrome-based coding, are suitable for variable-size fingerprint template. SS constructions for Hamming distance are further categorized into code-offset construction and syndrome construction.

Arakala et al. [127] proposed a set distance-based FE scheme which consists of both local and global quantization and representation of fingerprint minutiae. The local features used are based on nearest neighbour descriptor whereas the global quantization is performed in a polar coordinate system centred at the core point. Furthermore,

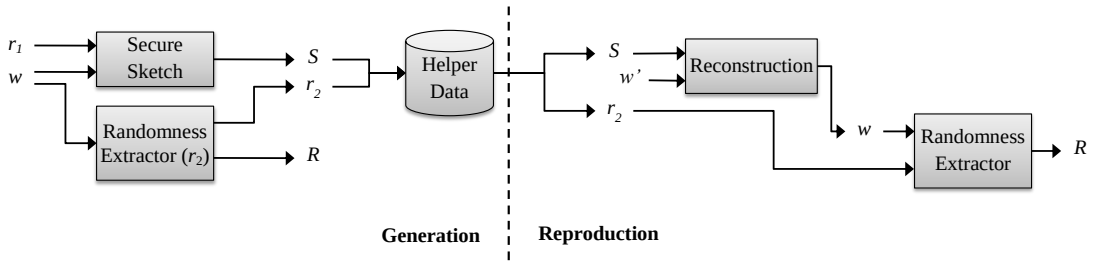


Figure 2.2: A general framework of FE [2]. In the diagram, w and $w\prime$ denotes the original and query biometric data respectively, r_1 denotes the randomly chosen codeword for sketch construction where applicable, r_2 is the randomness of the extractor, s is the sketch constructed and R represents the cryptographic key generated.

Arakala et al. [128] implemented SS using the Voronoi neighbors representation [62] of the fingerprint.

A SS based on code-offset construction [2] was proposed by Chang and Roy [129]. This method takes the minutiae counts on the left and right of numerous randomly drawn straight lines as the fingerprint representation. PCA is then applied to obtain the feature vector. The final bit-string is generated based on zero-thresholding. While using the same feature vector extraction method, Li et al. [130] performed components grouping and combination to improve the robustness of the system. The disadvantage of these methods is that they require the fingerprint images to be pre-aligned. Besides, the feature used is less discriminatory than previous examples [127, 128]. In order to achieve better accuracy, Liu et al. [131] proposed the combined feature-based sketch by fusing minutiae-based features and image-based features.

All the SS and FE implementations discussed above are based on the models proposed by Dodis et al. [2]. On the other hand, robust FE [132] and fully robust FE [133] were designed to increase the robustness of the cryptographic key generated by pre-authenticating the helper data during reconstruction.

Key-Binding Biometric Cryptosystems

As a contribution to key-binding biometric cryptosystems, Juels and Wattenberg [36] proposed a framework called fuzzy commitment. A random codeword from an error-correcting code is chosen. The hashed codeword and the difference between the codeword and the unprotected biometric template are stored as the helper data. Upon authentication, the recovered codeword is hashed and compared to the original hashed codeword. Tong et al. [134] demonstrated a practical use of the fuzzy commitment scheme on FingerCode [23] and showed a better result than using FingerCode alone. In addition, Nandakumar [135] implemented fuzzy commitment with minutiae phase spectrum [136, 137] to realize a fingerprint cryptosystem.

Another instance of key-binding biometric cryptosystem is the fuzzy vault scheme [138]. It is an order-invariant version of fuzzy commitment as it does not require the input biometric features to be ordered and fixed-length vector. The idea is to derive a polynomial from the minutiae set while adding some random chaff points which do not lie on the polynomial to conceal it. Error-correcting coding is used to reconstruct the original polynomial, provided that the query minutiae set is largely correlated with

the enrolled set. The number of chaff points added offers a trade-off between the security and the robustness of the biometric cryptosystem.

Clancy et al. [139] implemented the fuzzy vault scheme by directly using the Cartesian coordinates of the minutiae to generate the polynomial. Uludag et al. [140] employed the same fingerprint features in fuzzy vault, except that cyclic redundancy check was used instead of error-correcting coding due to difficulties of applying error-correcting on biometric data. However, fingerprint alignment is less discussed in the aforementioned literature. Uludag and Jain [141] used the invariant features of minutiae pairs for fingerprint alignment based on a voting system. The minutiae pairs between the enrolled set and the query set are matched and only possible correspondences generate votes for the relative alignment. Alignment using the most reliable minutiae pair has also been proposed [142]. Other fingerprint alignment techniques used with fuzzy vault include high curvature points-based technique [143], core point-based technique [144], alignment using geometric hash tables [145], alignment using local structures [146] and alignment based on minutiae orientation histograms [147].

On the other hand, an alignment-free fuzzy vault scheme was proposed by Li et al. [148], where orientation-based [49] and nearest neighbours-based minutia descriptor [39] were adapted. The first fuzzy vault is generated based on Huffman coding while the second vault is generated by polynomial construction using the two minutia descriptors respectively. Three score-level fusion strategies are used to provide the final matching decision.

Moreover, several attempts have been made to improve the accuracy and/or security of key-binding biometric cryptosystems such as password hardening [149], combination of fuzzy vault with fuzzy commitment [150,151], fuzzy vault using multiple polynomials [152], multi-biometric fuzzy vault and fuzzy commitment [153] and improved chaff points generation for fuzzy vault [154].

2.2.3 Hybrid BTP

In this thesis, we define hybrid BTP as the amalgamation of one of the models in cancellable biometrics with one of that in biometric cryptosystems to formulate a tangible solution towards BTP. With the intention to strengthen the security of fuzzy vaults against cross-matching of multiple vaults from the same user (linkability attack), Feng et al. [155] proposed an approach of incorporating cancellable fingerprint with fuzzy

vault scheme. The cancellable template is realized by applying a random Butterworth low pass filter on the fingerprint features. Another cancellable fuzzy vault scheme was presented by Xu and Wang [156], where the original minutiae are shifted based on a Gaussian function with random magnitude. Although these methods have improved the unlinkability of fuzzy vaults, the cancellable fingerprint schemes used can be easily reversed if the user-specific random number is known to the adversary.

Besides, Bringer et al. [157] proposed to apply secure sketches to cancellable fingerprint. A bit-string is extracted from the fingerprint based on Gabor response and reliable bits selection [158]. The revocability of the fingerprint template is realized via permutation of the bit-string. Finally, the sketch is a product code constructed from two binary Reed-Muller codes following a unique coding scheme described by Bringer et al. [159].

As literature of hybrid BTP using fingerprint is limited, we also look into its practicality in other biometric modalities. Using the concept of cancellable biometrics, Kanade et al. [160, 161] implemented a hybrid BTP on iris and face by shuffling the biometric template prior inputting it to a fuzzy commitment scheme. With the same shuffling technique applied on iris template, Fouad et al. [162] presented a cancellable fuzzy vault. Albeit revocable, the shuffling process is completely reversible if the shuffling key is compromised.

A more sophisticated hybrid BTP was demonstrated by Feng et al. [163] by combining cancellable face with fuzzy commitment scheme. The cancellable face template is generated through random projection and a unique template binarization method coined as the discriminability-preserving transform, of which the user-specific quantization thresholds are pre-trained with data clustering technique to maximize the discriminability of the resulting bit-string.

Additionally, Leng and Zhang [164] proposed a cancellable palmprint cryptosystem by embedding two-dimensional Palmprint Phasor that is similar to BioPhasor [126] into a key-binding biometric cryptosystem. Besides the random key for Palmprint Phasor generation, a second key is used for scrambling transformation [165, 166] to increase the randomness of the cancellable palmprint template.

2.2.4 Summary

Unless invariant features are used [99, 100, 104–106], non-invertible transforms by spatial perturbation requires pre-alignment of fingerprints [94–98, 101] and it usually implies loss in performance due to the difficulty in absolute pre-alignment. Besides, non-linear local distortions are not addressed in such schemes. On the contrary, permutation and RP used by biometric salting have been proven to be able to maintain the recognition performance of biometric templates with the downside of weaker security under stolen-key scenario. Therefore, incorporating non-invertible fingerprint feature extraction with biometric salting can be a promising solution towards the trade-off between performance and security.

For biometric cryptosystems, the key length is a major concern in the security measure [167]. It directly affects the entropy of the system and the probability of the secret key to be guessed. Another concern of a biometric cryptosystem is whether the helper data leaks critical information about the biometric data. Both the key length and the biometric data leakage are two important factors in determining the performance trade-off of a biometric cryptosystem. Besides, biometric cryptosystems which utilize error-correcting codes such as code-offset construction for SS and fuzzy commitment have low revocability as there are limited number of codewords can be chosen from the error-correcting codes. This leads to the exploration of hybrid BTP.

The existing hybrid BTP schemes for fingerprints [155, 156] have succeeded in employing variable-size cancellable template with fuzzy vault. As fuzzy vault is susceptible to linkability attack, the non-invertibility of the cancellable template chosen plays an important role in determining the security of the system. Unfortunately, none of the cancellable fingerprint generation techniques used in the existing hybrid schemes are safe in the case where the random key is compromised. The application of cancellable fingerprints, especially those with the nature of producing variable-size template, in key-generation biometric cryptosystems remains undiscovered.

CHAPTER 3

Multi-line Code: Minutiae-Based Cancellable Fingerprint Template

3.1 Introduction

Minutiae-based methods for fingerprint recognition utilize the ISO format of minutiae in the fingerprint to generate a biometric template. While features robustness and algorithm complexity are the two major concerns of designing a conventional fingerprint template, cancellable fingerprint template considers two additional properties — irreversibility and revocability, as discussed in section 1.3.

Inspired by Qi and Wang's [50] minutia descriptor constructed by taking samples from multiple lines centring at the reference minutia, we introduce a novel non-invertible minutia descriptor namely the multi-line code (MLC). However, instead of a texture-based descriptor, MLC describes a minutia based on fixed-radius vicinity at the sample points. On top of that, two revocation techniques, that is, permutation and random projection (RP), are used to realize the revocability property of the fingerprint template. In this chapter, the complete procedure of the cancellable template generation scheme is elaborated. The security and performance of the proposed scheme are also evaluated. Part of this work has been published [168, 169].

3.2 Nomenclature

Symbol	Description
$\mathbf{P}_{(i)} = [x_{(i)}, y_{(i)}, \theta_{(i)}]$	the ith minutia extracted from a fingerprint with its x-coordinate, y-coordinate and local ridge orientation
$\mathbf{P}_r = [x_r, y_r, \theta_r]$	a reference minutia for MLC generation
N_m	number of minutiae extractable from a fingerprint
$\varphi_{(i)}$	orientation difference between the reference minutia and the ith neighbouring minutia
N_φ	number of orientation levels in MLC construction
$\Delta\varphi$	quantization width of each orientation level
l	length of lines for MLC construction
N_l	number of lines for MLC construction
N_s	number of sample points taken for each line
θ_l	direction of a line in MLC generation
d	distance between two sample points
r	radius of circles centring at the sample points
$\boldsymbol{\omega} \in \mathbb{Z}_{\geq 0}^{D_m}$ or $\boldsymbol{\omega} \in \mathbb{R}_{\geq 0}^{D_m}$	minutia vector (MLC) of a minutia generated by either *MLCN* or *MLCD* algorithm
D_m	dimension of a MLC
D_s	dimension of a single-line code
$\boldsymbol{\Omega} \in \mathbb{Z}_{\geq 0}^{N_m \times D_m}$ or $\boldsymbol{\Omega} \in \mathbb{R}_{\geq 0}^{N_m \times D_m}$	MLC template of a fingerprint
$\mathbf{M}_\alpha$	direction mask for efficient MLC generation
$\hat{\mathbf{M}}_\alpha$	rotated direction mask
$\tilde{\mathbf{M}}_\alpha$	direction indicator mask
$\mathbf{P}_s = [x_s, y_s]$	coordinates of a sample point in MLC
$\mathbf{M}_\beta$	distance mask for efficient MLC generation
κ_c	user specific key for revocation
$\boldsymbol{\omega}'$	normalized minutia vector (MLC)
$\boldsymbol{\Omega}'$	normalized MLC template
$\hat{\boldsymbol{\Omega}} \in \mathbb{R}^{N_m \times D_r}$	cancellable MLC template

Symbol	Description
D_r	dimension of the minutia vectors of cancellable MLC template
$\mathbf{R} \in \mathbb{R}^{D_m \times D_r}$, $\mathbf{R}_\perp \in \mathbb{R}^{D_m \times D_r}$	random matrix and orthonormalized random matrix for RP
$\mathbf{S}_l$	local similarity matrix between two MLC templates
τ_s	local similarity threshold to determine if two minutiae are matchable
S_g	global similarity between two fingerprints

3.3 Multi-Line Code: Non-invertible Minutia Descriptor

3.3.1 Formulation of Multi-Line Code

Multi-line code (MLC) is a minutia descriptor constructed based on the spatial distribution of the neighbouring minutiae within a fixed radius. However, unlike some fixed-radius approaches [44, 45, 47], MLC does not create the circumference to only centre at the reference minutia itself but also at the sample points extended from the reference minutia. The formulation of MLC inspects the fingerprint in a three dimensional aspect which include the Cartesian plane (x-coordinate and y-coordinate) and the orientation dimension. Let $\{\mathbf{P}_{(i)}\}$ be the minutiae set extracted from the fingerprint, where $\mathbf{P}_{(i)} = [x_{(i)}, y_{(i)}, \theta_{(i)}]$ is an ISO representation of a minutia for $i \in [1, N_m]$ and N_m is the total number of minutiae extracted. Taking a reference minutia, $\mathbf{P}_r = [x_r, y_r, \theta_r]$ ($\mathbf{P}_r \in \{\mathbf{P}_{(i)}\}$) as an instance, the steps to generate a MLC are as follows:

1. Except for $\mathbf{P}_r$, other minutiae in the minutiae set $\{\mathbf{P}_{(i)}\}$ are quantized into N_φ levels according to the difference between the orientation of the minutiae and the orientation of the reference minutia as illustrated in Figure 3.1. The orientation difference, $\varphi_{(i)}$ is computed so that it ranges from 0 to 2π, as follow:

$$\varphi_{(i)} = \begin{cases} \theta_{(i)} - \theta_r, & \text{if } \theta_{(i)} \geq \theta_r; \\ 2\pi + (\theta_{(i)} - \theta_r), & \text{otherwise.} \end{cases} \tag{3.3.1}$$

Therefore, the quantization width for each orientation level is $\Delta\varphi = 2\pi/N_\varphi$.

2. At each orientation level, construct a straight line of length, l in the same direction as θ_r and take N_s sample points equally distributed along the line, separated by a distance, d with each other. Thus, the relationship between l, N_s and d is $N_s = l/d + 1$, of which d must be a factor of l.

3. A circle of radius, r is then drawn on the sample points marked in the previous step. For each circle, the *i*) number of minutiae; or the *ii*) mean of distances between the sample point (also the centre of the circle) and the minutiae, in the left and right semi-circles, separated by the straight line, are taken as the feature code. The two aforementioned MLC types are hereafter referred to as *MLCN* and *MLCD* respectively. The feature codes for all semi-circles on the line and for all orientation levels are concatenated to form a single-line code.

4. Repeat step 2 and step 3 for lines of different direction with equal angle in between each other. Suppose N_l lines are created, the possible directions of the lines, θ_l are θ_r, $\theta_r + \frac{\pi}{N_l}$, $\theta_r + \frac{2\pi}{N_l}$ ... $\theta_r + \frac{(N_l-1)\pi}{N_l}$ respectively. A MLC, which is also a minutia vector, is formed by concatenating the N_l single-line codes. Therefore, the minutia vector generated for $\mathbf{P}_r$ using MLC algorithm can be expressed as $\boldsymbol{\omega} \in \mathbb{Z}_{\geq 0}^{D_m}$ (for *MLCN*) or $\boldsymbol{\omega} \in \mathbb{R}_{\geq 0}^{D_m}$ (for *MLCD*), where $D_m = D_s N_l$ is the total dimension of the vector and $D_s = 2N_s N_\varphi$ is the length of a single-line code.

Finally, a MLC fingerprint template is obtained by iterating the steps elaborated above through all minutiae in $\{\mathbf{P}_{(i)}\}$ and can be represented as $\boldsymbol{\Omega} \in \mathbb{Z}_{\geq 0}^{N_m \times D_m}$ or $\boldsymbol{\Omega} \in \mathbb{R}_{\geq 0}^{N_m \times D_m}$. Since $\mathbb{Z}_{\geq 0}$ and $\mathbb{R}_{\geq 0}$ are subsets of $\mathbb{R}$ ($\mathbb{Z}_{\geq 0} \subset \mathbb{R}_{\geq 0} \subset \mathbb{R}$), a MLC template, let it be *MLCN* or *MLCD*, is henceforth generalized as $\boldsymbol{\Omega} \in \mathbb{R}^{N_m \times D_m}$ for the convenience of denotations.

Since the straight lines constructed are derived from the position and orientation of the reference minutia, the code is computed based on the relative location and angle between the reference minutia and the neighbouring minutiae. In this way, MLC is made alignment-free as the codes extracted are invariant to global translation and rotation. Although the MLC algorithm is not completely scale-invariant, the parameters can be adjusted so that it is less sensitive to scaling, e.g. the radius of the circles must be large enough to address local translation of minutiae due to linear scaling. This also applies to perturbation of minutiae caused by local non-linear distortions. In addition, unlike nearest neighbour-based minutia descriptors where missing or spurious minutiae may

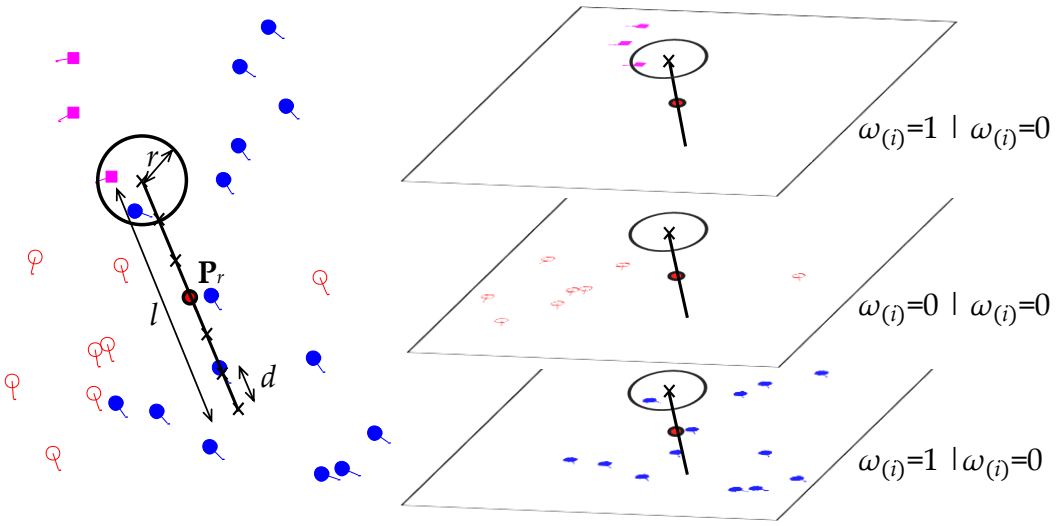


Figure 3.1: An illustration of the formulation of *MLCN*. The parameters (l, d and r) are labelled accordingly. The orientation levels which the minutiae belong to are differentiated by the markers of the minutiae. The graphic shows a straight line drawn across the reference minutia ($\mathbf{P}_r$) and a circle drawn centring at one of the marked sample points. In each orientation level, the number of minutiae in the left and right semi-circles ($\omega_{(i)}$) are taken as the feature code.

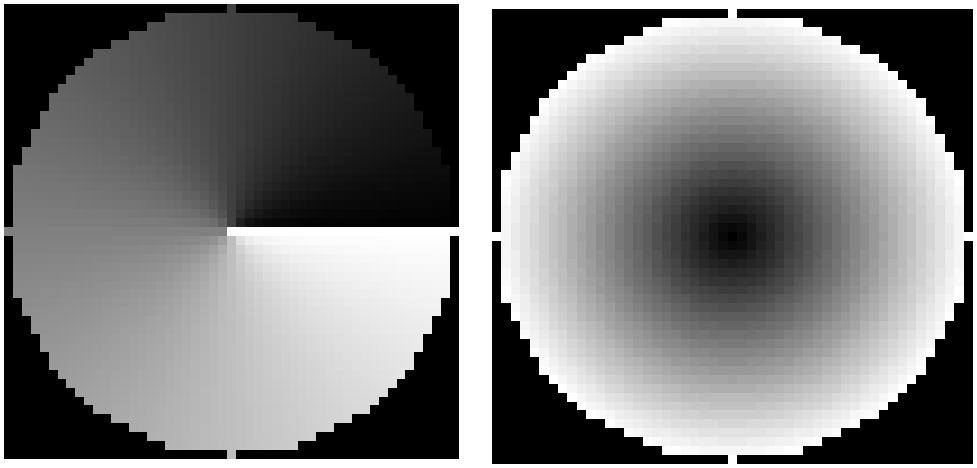
significantly alter the output minutia vector, the fixed radius-based minutia descriptor used is more robust against missing and spurious minutiae as the effect may be averaged by other minutiae within the radius.

3.3.2 Efficient MLC Generation

A direct approach of finding neighbouring minutiae that are within a fixed radius, r from a sample point is to evaluate the distance of every minutiae in the fingerprint from the sample point and select those with distance less than r. The process involves repetitive and redundant calculations for point-to-point distance, hence is inefficient. In this thesis, we suggest a simplified method of finding neighbouring minutiae, specially designed for MLC generation, which eliminates unnecessary calculations and consequently, reduces computational complexity of the entire algorithm.

The proposed simplified method utilizes a pre-defined mask to obtain the number of minutiae (*MLCN*) within the two semi-circles centring at a sample point. The pre-defined mask, also known as the direction mask, is a $(2r+1) \times (2r+1)$ matrix, $\mathbf{M}_\alpha$, in which the elements with distance more than r from the central element are assigned with 0's while the value of the remaining elements are determined by their direction from the central element in the four quadrants basis, ranging from 0 to 2π, i.e. $\mathbf{M}_\alpha \in$

$\{\mathbb{R} \cap [0, 2\pi]\}^{(2r+1)\times(2r+1)}$. The elements at zero degree are assigned with the value 2π so that they are distinguishable from the elements outside the circumference. An example of the direction mask is shown in Figure 3.2a.



(a) A 51×51 direction mask used for MLC generation. The angle values ranging from 0 to 2π are represented in grayscale with 0 and 2π corresponding to pure black and pure white respectively.

(b) A 51×51 distance mask used for MLC generation. The distance values ranging from 0 to 25 are visualized in full grayscale colormap.

Figure 3.2: Two types of pre-defined masks used for simplified generation scheme of MLC when $r = 25$.

Given the ISO template of a minutiae set, N_φ minutiae maps can be constructed. Each of these minutiae maps corresponds to the minutiae in different orientation levels. They are essentially binary matrices with the size same as that of the fingerprint image and with 1's indicating the locations of the minutiae. Figure 3.3 shows the graphical illustration of obtaining the number of minutiae within two semi-circles given the pre-defined direction mask ($\mathbf{M}_\alpha$), a minutiae map, the position of a sample point ($\mathbf{P}_s = [x_s, y_s]$) and the direction of the straight line (θ_l). First, subtract θ_l from $\mathbf{M}_\alpha$ to get the rotated direction mask, $\hat{\mathbf{M}}_\alpha = \mathbf{M}_\alpha - \theta_l$. Then, the mask is converted into an indicator matrix, $\tilde{\mathbf{M}}_\alpha \in \{1, -1\}^{(2r+1)\times(2r+1)}$ with 1's indicating the left semi-circle area and -1's the right semi-circle area following the conditions below:

$$\tilde{M}_{\alpha(ij)} = \begin{cases} 1, & \text{if } 0 \leq \hat{M}_{\alpha(ij)} < \pi; \\ -1, & \text{otherwise,} \end{cases} \tag{3.3.2}$$

where $\tilde{M}_{\alpha(ij)}$ and $\hat{M}_{\alpha(ij)}$ are the element at the ith row and jth column in $\tilde{\mathbf{M}}_\alpha$ and $\hat{\mathbf{M}}_\alpha$ respectively.

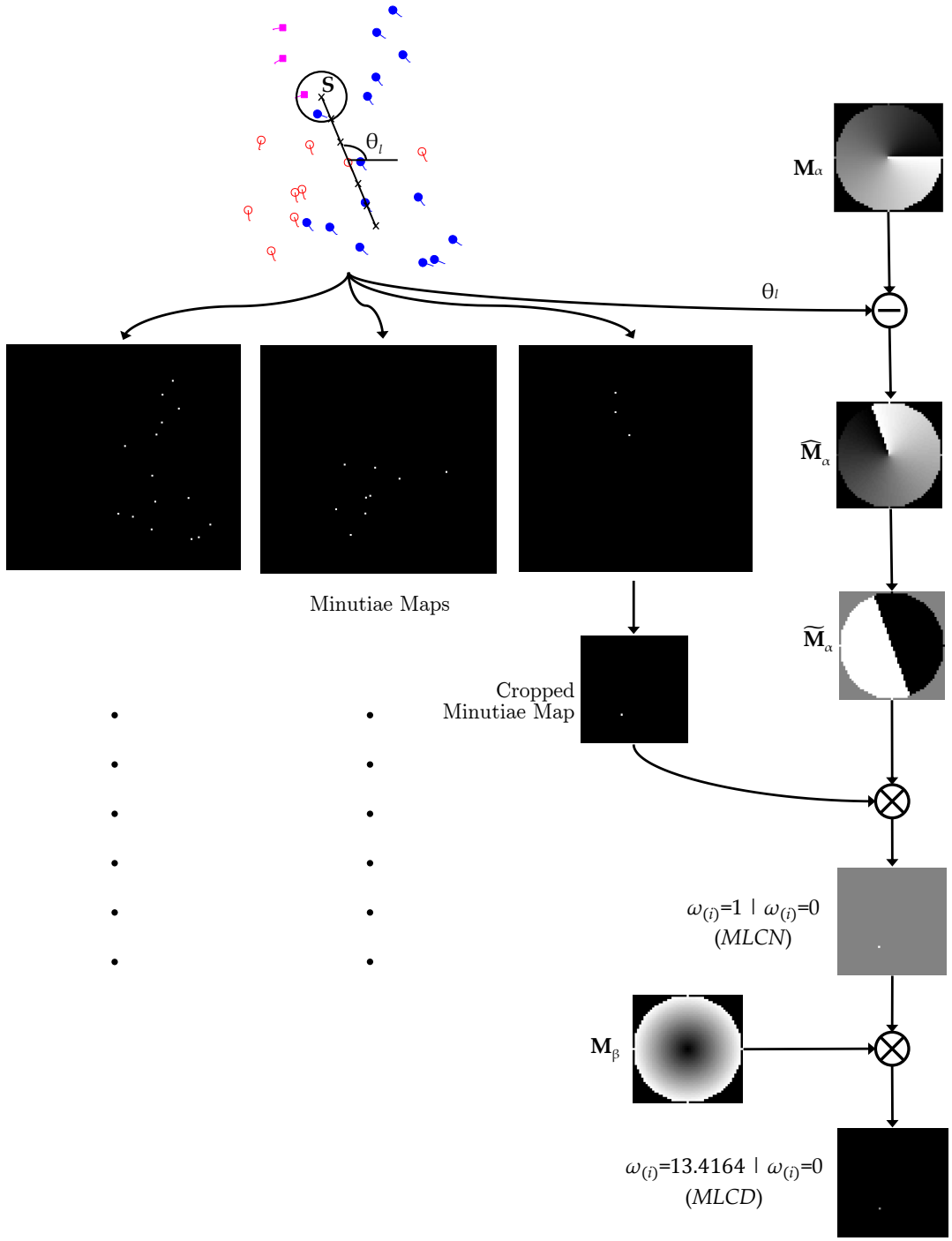


Figure 3.3: Masking technique to obtain MLC feature code at a given sample point.

Finally, a $(2r+1) \times (2r+1)$ area from the minutiae map centring at (x_s, y_s) is cropped out and element-wise multiplied with the indicator matrix ($\tilde{\mathbf{M}}_\alpha$). If the cropped area exceeds the boundaries of the minutiae map, 0's are padded onto the missing part. The number of 1's and -1's in the resulting matrix represents the number of minutiae on the left and right semi-circles.

For *MLCD* however, an additional distance mask, $\mathbf{M}_\beta \in \mathbb{R}^{(2r+1)\times(2r+1)}$, is required to obtain the distance of the minutiae within the semi-circles from the sample point. The value of the elements in $\mathbf{M}_\beta$ is equivalent to its distance from the central element, except that the elements falling outside the circumference are assigned 0's, as shown in Figure 3.2b. In order to get the *MLCD* code, the cropped minutiae map is element-wise multiplied with both $\tilde{\mathbf{M}}_\alpha$ and $\mathbf{M}_\beta$. The mean of positive distances in the resulting matrix is the code for the left semi-circle, whereas the mean of negative distances contributes to the right semi-circle.

3.4 Cancellable Template Generation

In this thesis, we utilize two distinct techniques to generate revocable MLC template, namely permutation and random projection (RP). Revocable transformation of MLC template is performed at the minutia vector level via a user-specific key, κ_c. In the case of permutation-based transformation, κ_c is used to generate a random permutation order. Since MLC template is unordered and variable-size, the randomly generated permutation order is universal to all minutia vectors. The minutiae vectors are re-arranged according to the new order and the resulting cancellable MLC template is denoted as $\hat{\mathbf{\Omega}} \in \mathbb{R}^{N_m \times D_r}$, where $D_r = D_m$ in the case of permutation-generated templates.

On the other hand, RP-based transformation uses κ_c as the seed to generate a pseudo-random projection matrix following the steps below:

1. Use κ_c to generate a set of pseudo-random numbers $\mathbf{R} \in \mathbb{R}^{D_m \times D_r}$ from the $\mathcal{N}(0,1)$ distribution, where D_r is the desired output dimension and $D_r \leq D_m$.

2. Apply the Gram-Schmidt process (see Appendix A) to orthonormalize $\mathbf{R}$ into $\mathbf{R}_\perp \in \mathbb{R}^{D_m \times D_r}$ so that $\mathbf{R}_\perp \mathbf{R}_\perp^{\mathrm{T}} = \mathbf{I}$ and $\|\mathbf{R}_\perp\|_2 = 1$, where $\mathbf{I}$ is the identity matrix and $\|\cdot\|_2$ denotes the ℓ^2 norm.

Prior the transformation, the minutiae vectors in the original MLC template ($\mathbf{\Omega}$) is normalized, resulting in $\mathbf{\Omega}'$, with $\boldsymbol{\omega}'_{(i)} = \boldsymbol{\omega}_{(i)} / \|\mathbf{\Omega}\|_2$ denoting the ith normalized minutia vector. The final RP-based cancellable MLC template is simply calculated as

$$\hat{\mathbf{\Omega}} = \mathbf{\Omega}' \mathbf{R}_\perp, \tag{3.4.1}$$

with $\hat{\mathbf{\Omega}} \in \mathbb{R}^{N_m \times D_r}$.

3.5 MLC Template Matching Algorithm

Unlike fixed-length representations of which one template can be directly compared with another template using desired metrics, the matching of MLC templates involve two phases, viz. local and global matching. Local matching compares the minutia vectors among two templates and find the matchable minutiae pairs; whereas global matching uses the matchable pairs found to conclude the similarity of two fingerprint templates. The details of each matching phase are explained in this section and is summarized in algorithm 3.1.

Algorithm 3.1: MLC template matching

Function *MLCMatch*($\hat{\mathbf{\Omega}}_E, \hat{\mathbf{\Omega}}_Q, \tau_s$)
  $N_{mE} \longleftarrow \text{size}(\hat{\mathbf{\Omega}}_E)$
  $N_{mQ} \longleftarrow \text{size}(\hat{\mathbf{\Omega}}_Q)$
  $\mathbf{S}_l, \mathbf{A} \longleftarrow 0$
  for $i \in [1, N_{mE}]$ **do**
    $MLC1 \longleftarrow \hat{\mathbf{\Omega}}_E[i]$ // $\hat{\boldsymbol{\omega}}_{E(i)}$
    for $j \leftarrow 1, N_m^Q$ **do**
      $MLC2 \longleftarrow \hat{\mathbf{\Omega}}_Q[j]$ // $\hat{\boldsymbol{\omega}}_{Q(j)}$
      $\mathbf{S}_l[i,j] \longleftarrow \text{Dice}(MLC1, MLC2)$ `// local matching, refer to` (3.5.1)
    end
  end
  `/* local similarities filtering, refer to` (3.5.2) `*/`
  enrolledMax $\longleftarrow \max(\mathbf{S}_l, 2)$
  queryMax $\longleftarrow \max(\mathbf{S}_l, 1)$
  for $i \in [1, N_{mE}]$ **do**
    for $j \in [1, N_{mQ}]$ **do**
      $\mathbf{A}[i,j] \longleftarrow (\textbf{enrolledMax}[i] == \textbf{queryMax}[j])$
    end
  end
  $\mathbf{S}_l \longleftarrow \mathbf{S}_l .^* \mathbf{A}$
  $S_g \longleftarrow \text{sum}(\mathbf{S}_l)/\min(N_{mE}, N_{mQ})$ `// global matching, refer to` (3.5.3)
  return S_g
end

3.5.1 Local Matching

Given a minutia vector, $\hat{\boldsymbol{\omega}}_{E(i)}$ taken from the enrolled fingerprint template ($\hat{\mathbf{\Omega}}_E \in \mathbb{R}^{D_r \times N_{mE}}$) and a minutia vector, $\hat{\boldsymbol{\omega}}_{Q(j)}$ taken from the query fingerprint template ($\hat{\mathbf{\Omega}}_Q \in \mathbb{R}^{D_r \times N_{mQ}}$), where N_{mE} and N_{mQ} are the number of minutiae in $\hat{\mathbf{\Omega}}_E$ and $\hat{\mathbf{\Omega}}_Q$ respectively.

The similarity between $\hat{\boldsymbol{\omega}}_{E(i)}$ and $\hat{\boldsymbol{\omega}}_{Q(j)}$ signifies the likelihood of $\hat{\boldsymbol{\omega}}_{E(i)}$ in $\hat{\boldsymbol{\Omega}}_E$ being a correspondence to $\hat{\boldsymbol{\omega}}_{Q(j)}$ in $\hat{\boldsymbol{\Omega}}_Q$.

In this thesis, we use Dice's coefficient as the similarity measure between two minutia vectors, calculated as

$$S_{l(ij)} = \frac{2\langle \boldsymbol{\omega}_{E(i)}, \boldsymbol{\omega}_{Q(j)} \rangle}{\|\boldsymbol{\omega}_{E(i)}\|_2^2 + \|\boldsymbol{\omega}_{Q(j)}\|_2^2} \tag{3.5.1}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product between two vectors. The value of $S_{l(ij)}$ ranges from 0 to 1, of which 0 indicates a total mismatch between $\boldsymbol{\omega}_{E(i)}$ and $\boldsymbol{\omega}_{Q(j)}$, whereas 1 indicates a perfect match between them.

3.5.2 Global Matching

When dealing with two fingerprint templates, each minutiae vector in $\hat{\boldsymbol{\Omega}}_Q$ is cross-matched with the ones in $\hat{\boldsymbol{\Omega}}_E$ so that we have a similarity matrix containing similarity scores among all minutia vectors between two templates. Each element in the similarity matrix is then re-evaluated with the following criterion to eliminate double-matching:

$$S_{l(ij)} = \begin{cases} S_{l(ij)} & \text{Condition 1;} \\ 0 & \text{otherwise.} \end{cases} \tag{3.5.2}$$

Condition 1 implies that $S_{l(ij)} \geq \tau_s$ and $S_{l(ij)}$ must be the maximum among all values of $S_{l(ej)}$ (for $e \in [1, N_{mQ}]$) and $S_{l(iq)}$ (for $q \in [1, N_{mE}]$), that is, $S_{l(ij)}$ must be the maximum among the elements in its row and column. τ_s is the lowest similarity threshold to conclude that a pair of minutia vectors are matchable.

To determine the overall similarity between $\hat{\boldsymbol{\Omega}}_E$ and $\hat{\boldsymbol{\Omega}}_Q$, a global matching score is used to measure the likelihood of them being two instances of the same fingerprint. With the processed similarity matrix, we calculate the matching score with the following formula:

$$S_g = \frac{\sum_{i=1}^{N_{mE}} \sum_{j=1}^{N_{mQ}} S_{l(ij)}}{\min(N_{mE}, N_{mQ})} \tag{3.5.3}$$

3.6 Experiments and Analyses

3.6.1 Datasets and Testing Protocol

Four public fingerprint datasets including FVC2002 DB1, FVC2002 DB2 [170], FVC2004 DB1 and FVC2004 DB2 [171] are used to examine the proposed MLC in this thesis. FVC2004 datasets yields higher difficulty than FVC2002 datasets due to exaggerated perturbations such as non-linear distortion, dryness and wetness. The minutiae are extracted from the fingerprint images via VeriFinger SDK [172] so that the results obtained are comparable to the results in the literature [1]. Some statistical information of the fingerprint datasets are tabulated in Table 3.2.

Table 3.2: Statistical information of the fingerprint datasets used for experiments.

Dataset	Image dimensions (in pixels)	Average ROI dimensions (in pixels)	Average number of minutiae extracted
FVC2002 DB1	388×374	196×278	33
FVC2002 DB2	296×560	229×402	42
FVC2004 DB1	640×480	200×318	38
FVC2004 DB2	328×364	200×277	33

Each of the datasets consists of fingerprints from 100 users with 8 samples per user. In the experiments, the first sample of every user's fingerprint is used as the enrolled template while the remaining samples are treated as queries. As a result, there are 700 genuine matchings and 69300 impostor matchings for each dataset. Furthermore, since the RP-based experiment in section 3.6.3 involves randomly generated matrix, the experiment was executed ten-fold with different random seed assigned to each user in every trial to obtain an average measure of the performance.

3.6.2 MLC Parameters Tuning

In this section, we perform exhaustive search through multiple MLC parameters combinations to determine the best-performing parameters for each dataset. There are three manipulated parameters in the formulation of MLC, viz. the length of the straight lines, l, the distance between two sample points, d and the radius of the circles, r, while number of orientation levels, N_φ and the number of lines, N_l are fixed to 6 and 3 respectively. Three l values are chosen inclusive of 240, 280 and 320, while r ranges from 15 to 30

[1] Note that minutiae extraction in our previous publications [168, 169] follows a different method [173], hence the results shown are different.

Table 3.3: Length of the minutia vectors produced by MLC (both *MLCN* and *MLCD*) with different parameter values.

l	d	D_m
240	6	1476
	8	1116
	10	900
	12	756
	15	612
	16	576
	20	468
	24	396
280	7	1476
	8	1296
	10	1044
	14	756
	20	540
	28	396
320	8	1476
	10	1188
	16	756
	20	612
	32	396

with a step size of 5. The values of d are chosen so that they are factors of l and that they range from $\frac{l}{40}$ to $\frac{l}{10}$. The resulting length of minutia vectors (D_m) with different parameters are shown in Table 3.3.

Tables 3.4 to 3.7 show the performance of MLC applied on four different fingerprint datasets. The best EERs for *MLCN* are 2.25%, 1.94%, 8.20% and 8.05%, whereas the lowest EERs achieved by *MLCD* are 2.32%, 1.65%, 7.54% and 7.82% for FVC2002 DB1, FVC2002 DB2, FVC2004 DB1 and FVC2004 DB2 respectively. Overall, *MLCD* performs slightly better than *MLCN*. *MLCN* records the number of minutiae in the semi-circles regardless of the position of the minutiae, while *MLCD* renounces the information about the number of minutiae and describes the average distance of the minutiae from the centre. The results show that average distance is a more distinctive feature than minutiae count for a fixed-radius descriptor.

The performance shows no linear relationship with any one parameter but is correlated with all parameters. From the aspect of r, the EERs are lower at $r = 20$ and $r = 25$. This is because when r is too small ($r = 15$), the semi-circles formed are not sufficient for capturing the neighbouring minutiae at the sample points. While the semi-circles created are meant for capturing local information, excessively large r ($r = 30$) introduces high

Table 3.4: EER (in %) of MLC using FVC2002 DB1.

l	*d*	*MLCN*				*MLCD*			
		r				*r*			
		15	20	25	30	15	20	25	30
240	6	3.69	2.54	2.38	3.00	3.61	2.55	**2.32**	2.38
	8	3.49	2.94	2.56	3.19	3.51	3.16	2.47	2.35
	10	4.39	2.77	**2.25**	2.80	4.75	4.09	2.60	2.59
	12	5.14	2.76	2.39	3.03	6.84	3.19	2.60	2.52
	15	4.27	2.85	2.29	3.28	8.20	3.88	2.76	2.87
	16	6.05	3.24	2.88	3.29	5.28	3.39	3.64	2.37
	20	5.28	2.94	3.27	2.99	5.03	5.72	3.15	2.71
	24	6.39	2.82	3.63	3.75	5.27	3.74	5.92	3.62
280	7	3.97	2.68	2.86	3.19	5.17	3.14	2.55	2.68
	8	4.39	3.09	2.71	3.14	4.19	3.77	2.86	2.64
	10	4.01	2.86	2.50	2.96	4.36	3.92	2.54	2.72
	14	5.65	2.78	3.15	3.00	9.44	2.81	3.09	3.05
	20	4.33	3.00	3.14	3.32	4.93	5.86	2.95	2.95
	28	7.09	3.59	2.57	4.16	6.77	3.75	3.76	4.64
320	8	3.22	3.07	3.05	3.30	3.50	3.10	2.99	2.78
	10	4.01	2.97	2.71	2.97	4.19	4.20	2.81	2.67
	16	3.83	2.78	2.73	3.23	4.37	3.76	3.05	2.65
	20	4.58	3.03	3.02	3.41	4.89	5.80	3.18	2.73
	32	7.86	3.45	3.07	3.60	7.44	4.30	4.13	3.82

Table 3.5: EER (in %) of MLC using FVC2002 DB2.

l	*d*	*MLCN*				*MLCD*			
		r				*r*			
		15	20	25	30	15	20	25	30
240	6	3.29	2.22	2.01	2.52	4.26	2.38	1.88	1.79
	8	3.62	2.21	2.06	2.48	3.20	2.55	1.81	1.86
	10	3.82	2.85	2.10	2.50	4.57	3.04	2.04	1.59
	12	4.67	2.39	2.49	2.34	6.65	2.33	2.28	2.00
	15	5.22	2.94	2.43	2.69	10.27	3.37	1.99	1.94
	16	6.02	2.96	2.96	2.50	5.82	2.29	3.20	1.71
	20	5.51	2.70	2.73	2.68	3.72	5.90	3.20	1.99
	24	7.08	3.00	3.36	2.95	5.45	2.76	4.91	2.76
280	7	3.38	2.19	2.32	2.67	4.26	2.05	2.10	1.94
	8	4.12	2.39	2.29	2.85	4.44	2.82	1.95	1.86
	10	3.44	2.68	2.00	2.71	4.26	3.06	1.95	1.71
	14	4.95	2.68	2.41	2.71	8.23	2.52	1.91	1.81
	20	5.27	2.76	2.63	2.59	3.42	5.53	3.21	2.10
	28	8.15	3.25	2.72	3.30	5.18	3.02	2.37	3.98
320	8	3.24	2.74	**1.94**	2.84	3.74	2.42	2.07	1.89
	10	3.52	2.58	2.16	2.86	4.35	3.25	**1.65**	1.98
	16	3.81	2.75	2.33	2.64	3.56	3.75	2.01	2.19
	20	4.40	2.69	2.83	3.07	3.31	5.29	2.85	1.98
	32	8.34	3.10	2.58	3.04	6.08	3.78	2.88	2.29

Table 3.6: EER (in %) of MLC using FVC2004 DB1.

l	*d*	*MLCN*				*MLCD*			
		r				*r*			
		15	20	25	30	15	20	25	30
240	6	12.97	9.24	**8.20**	8.58	13.82	10.15	8.96	7.79
	8	12.25	9.79	8.60	8.33	13.93	10.89	8.64	**7.54**
	10	13.45	10.53	8.64	8.85	14.90	12.19	9.52	8.33
	12	14.66	10.01	9.18	9.11	16.82	10.40	9.42	8.29
	15	14.26	10.68	9.38	8.86	19.06	12.12	9.45	8.76
	16	15.53	10.61	9.36	8.72	15.25	11.13	9.88	8.48
	20	15.43	11.69	8.93	8.91	16.11	14.60	10.56	9.05
	24	16.64	11.20	10.33	9.54	17.05	12.86	13.43	9.20
280	7	11.63	9.75	8.99	8.51	15.41	10.48	9.33	7.60
	8	13.54	10.26	9.02	8.59	15.23	11.57	9.04	7.69
	10	13.28	9.98	9.23	8.95	15.11	12.66	9.70	8.79
	14	15.03	9.62	9.29	8.88	18.07	11.05	10.35	8.45
	20	14.11	11.37	9.23	8.99	16.29	15.72	11.05	8.26
	28	18.31	11.61	9.45	10.19	17.14	13.26	11.39	11.10
320	8	12.21	9.73	9.29	9.63	14.11	10.38	9.24	8.78
	10	13.32	10.64	8.91	9.67	15.38	12.35	9.67	9.05
	16	13.14	10.85	9.53	9.42	15.63	11.65	10.03	9.37
	20	14.50	11.23	8.88	9.08	16.30	15.45	10.13	8.76
	32	19.04	11.54	10.46	9.97	19.19	13.19	11.96	10.34

Table 3.7: EER (in %) of MLC using FVC2004 DB2.

l	*d*	*MLCN*				*MLCD*			
		r				*r*			
		15	20	25	30	15	20	25	30
240	6	11.44	9.22	8.99	9.03	12.63	9.46	8.18	7.91
	8	11.77	9.25	8.65	9.01	11.21	9.14	8.49	7.95
	10	12.05	8.62	8.64	8.32	13.41	10.49	8.10	**7.82**
	12	13.06	9.43	9.62	8.44	14.74	9.24	9.14	8.31
	15	12.77	10.77	9.37	9.08	18.24	10.34	7.94	8.19
	16	15.03	9.34	8.77	9.20	14.80	9.02	10.36	8.76
	20	14.08	9.33	9.96	9.05	13.10	13.65	10.62	8.56
	24	15.77	10.07	11.46	9.27	13.73	10.22	13.48	9.62
280	7	11.94	8.66	**8.05**	8.61	13.85	8.89	8.40	8.02
	8	12.13	8.74	8.69	8.61	13.91	10.28	8.23	7.96
	10	11.78	8.70	8.36	8.25	13.09	10.43	8.06	8.41
	14	14.34	9.82	8.07	9.60	18.39	9.81	8.64	8.83
	20	13.15	9.64	9.62	9.20	13.74	13.62	10.58	8.64
	28	17.38	10.46	9.69	10.01	14.77	11.59	9.12	11.21
320	8	11.04	9.10	8.86	8.71	11.54	9.50	8.81	8.43
	10	11.78	9.02	8.47	8.54	13.23	10.68	8.84	8.31
	16	11.33	9.62	8.63	9.35	11.73	11.37	9.05	8.40
	20	13.11	9.56	9.30	9.19	14.10	13.69	10.66	8.59
	32	18.76	11.68	9.87	9.34	13.95	11.40	10.04	9.11

error tolerance and causes globalization of the minutiae neighbourhood. Therefore, an adequate r value balances between the two scenarios so that the EER is minimized. On the other hand, a minutiae neighbourhood is acquired in every d distance along the lines. An extremely small d value may result in redundancy, whereas some useful information along the line may be missed out when d is too large.

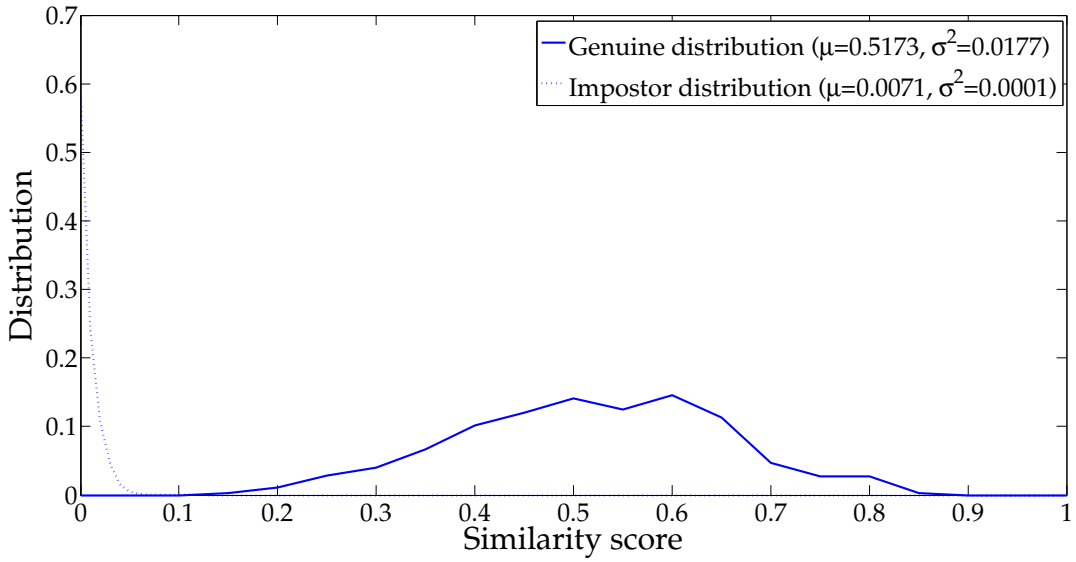
Furthermore, l is another factor affecting the performance of MLC. The best-suited l value for each dataset varies depending on the average ROI area of the dataset as stated in Table 3.2. As a semi-circle containing no minutiae is assigned the code '0' (for both *MLCN* and *MLCD*), it is redundant to extend the lines outside the ROI where minutiae do not exist. With $l = 320$, MLC yields the best performance for FVC2002 DB2 due to its relatively larger average ROI area as compared to the other datasets which peak at $l = 240$ or $l = 280$. In practice, the parameters are dependent on the fingerprint sensor type, the image size and the image resolution, thus a change in one of these factors requires the retraining of the optimal parameters (l, d and r).

3.6.3 Performance of Cancellable MLC

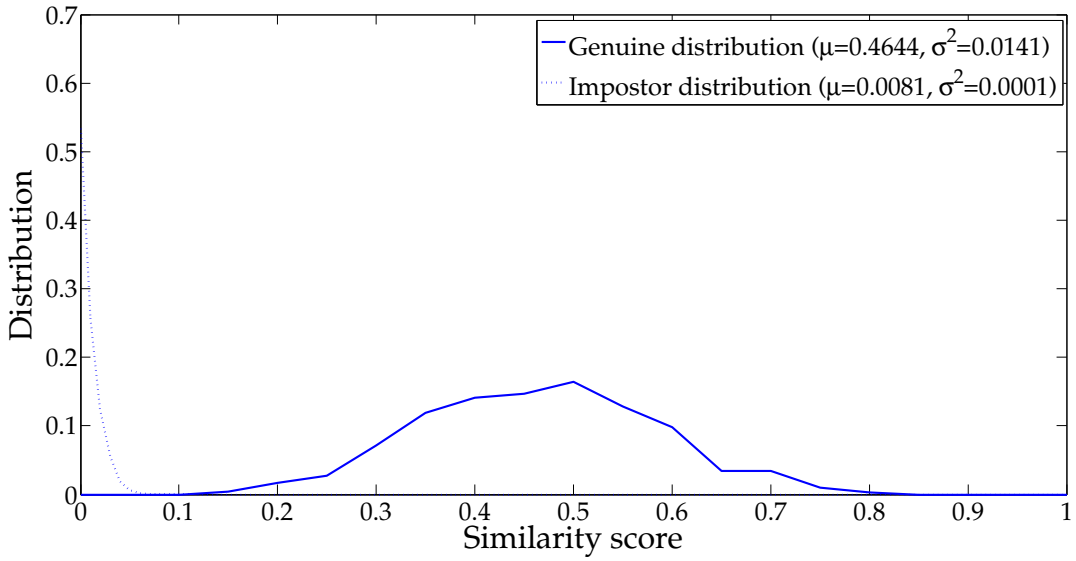
Teoh et al. [110] suggested that the user-specific key, κ_c used for template revocation can be tokenized and kept by the user. Therefore, the performance of cancellable MLC may be evaluated in two scenarios: *i*) the genuine key scenario, in which the impostor uses his own key to generate the query template; and *ii*) the stolen key scenario, in which the impostor has the knowledge of the key of the user being imposted and the query template is generated based on the same key, hence is also known as the same key scenario.

The optimal EER obtained for the genuine key scenario for both permutation-based and RP-based cancellable MLC are 0%. This indicates that all users can be correctly verified and no impostor will be falsely accepted under the scenario given an appropriate threshold for the matching score. For instance, figure 3.4 depicts the zero over-lapping area between the genuine distribution and the impostor distribution of RP-based cancellable MLC when $D_r = 100$ under genuine key scenario.

Considering the stolen key scenario, since permutation merely change the positions of the minutia vector components without altering the values, the performance is identical to that of without revocable transformation as discussed in section 3.6.2.



(a) *MLCN*



(b) *MLCD*

Figure 3.4: Genuine key score distribution of RP-based cancellable MLC with $D_r = 100$ for FVC2002 DB1. The genuine distribution and the impostor distribution are well-separated, with operational threshold at approximately 0.1 for both *MLCN* and *MLCD*.

On the other hand, the performance of RP-based revocable template depends on the MLC dimension after RP, D_r. Figure 3.5 shows the reaction of the system EER versus the reduced dimension for all datasets and for both *MLCN* and *MLCD*. In the figure, the EER deterioration ratio is defined as:

$$\text{ratio}_{\text{EER}} = \frac{\text{EER after RP}}{\text{EER before RP}}. \tag{3.6.1}$$

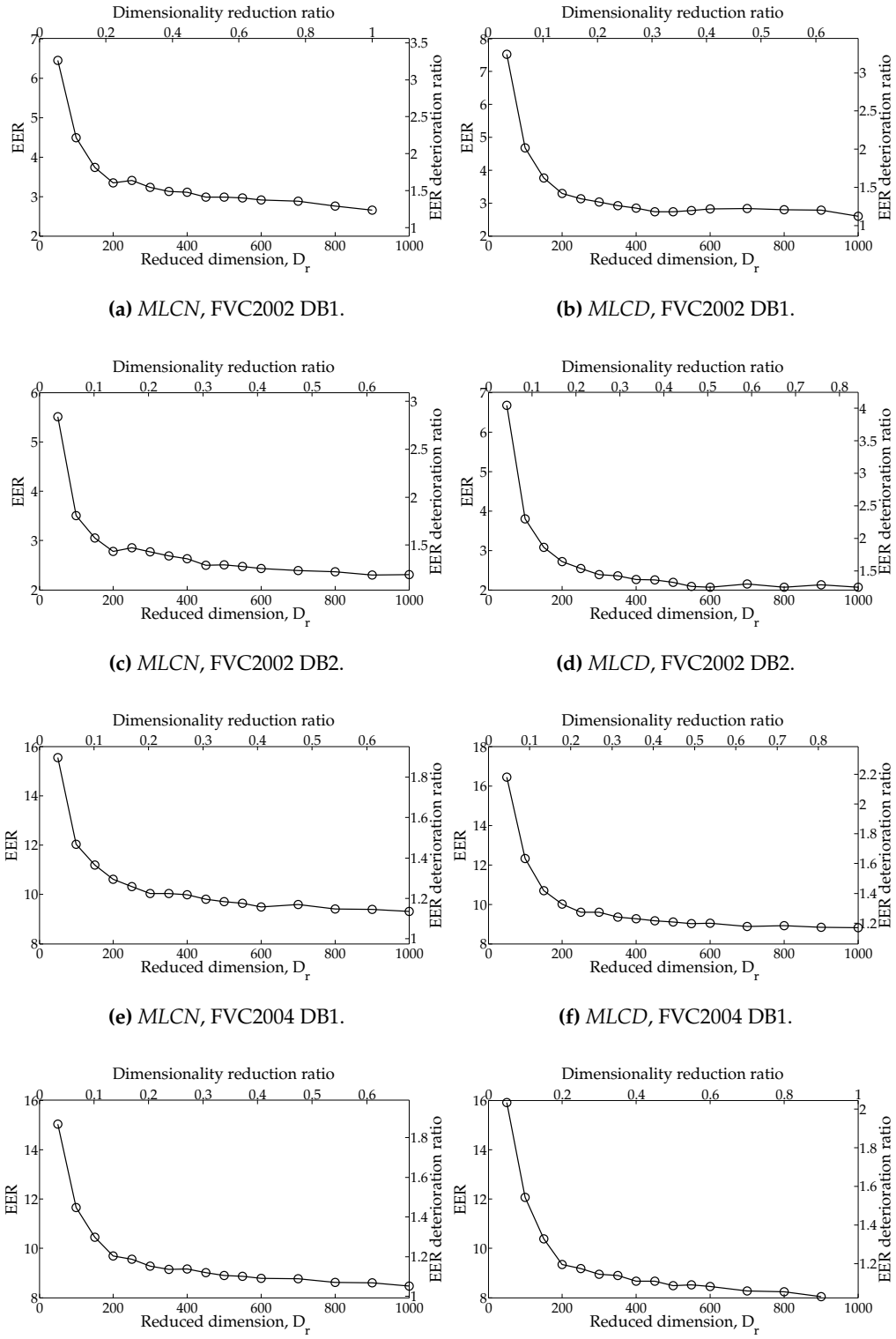
The EER deterioration of 1 indicates that the performance is well-preserved after RP; otherwise the ratio is larger than 1 when the performance deteriorates. A ratio of less than 1 is not possible as there is bound to have loss of information in the MLC after RP. Recall that the purpose of RP is for template revocation rather than to improve the performance of MLC, and this experiment aims at finding the reduced dimension which minimizes the loss of information. Besides the EER deterioration ratio, the dimensionality reduction ratio is defined as:

$$\text{ratio}_{\text{DR}} = \frac{D_r}{D_m}. \tag{3.6.2}$$

From figure 3.5, one can observe that as D_r increases from the range of 50 to 1000, the EER after RP converges to the original EER of MLC. The EER deterioration ratio is able to drop below 1.5 when $\text{ratio}_{\text{DR}} \geq 0.4$ for all cases. Although the performance continues to improve after that, the margin is insignificant. Neither *MLCN* nor *MLCD* can preserve all of the information in MLC after RP (i.e. reaching $\text{ratio}_{\text{EER}} = 1$) within the dimension range tested. This is not desirable, but it agrees with the results obtained for FaceHashing [111].

It has been discussed [174] that data clusters can be well-separated after RP if the original data is Gaussian or mixture of Gaussians. Also, Bingham and Mannila [175] has shown that RP can preserve the information in image and text data, which are generally Gaussian. In the case of MLC however, the minutia vectors are sparse in nature as most semi-circles contain no minutia. While such data is projected onto a lower dimensional Gaussian space, performance preservation is not guaranteed. Therefore, we can conclude that the performance of cancellable MLC is slightly worse than the original MLC under the circumstance that the impostor gets hold of the user's key. Figure 3.6 shows the comparison between the genuine-impostor distribution of the original MLC and the RP-based cancellable MLC. The distributions of the cancellable MLC are closer to each other compared to the original MLC, resulting in a larger over-lapping area. This can be verified by the slight decrease in the mean of genuine scores and increase in the mean of impostor scores.

Table 3.8 summarizes the performance of cancellable MLC under stolen-key scenario with while compared to other existing cancellable fingerprint generation methods. The recognition accuracy of the proposed cancellable MLC is comparable to other methods and is significantly better when more difficult datasets (FVC2004 datasets) are used. It



(a) *MLCN*, FVC2002 DB1.

(b) *MLCD*, FVC2002 DB1.

(c) *MLCN*, FVC2002 DB2.

(d) *MLCD*, FVC2002 DB2.

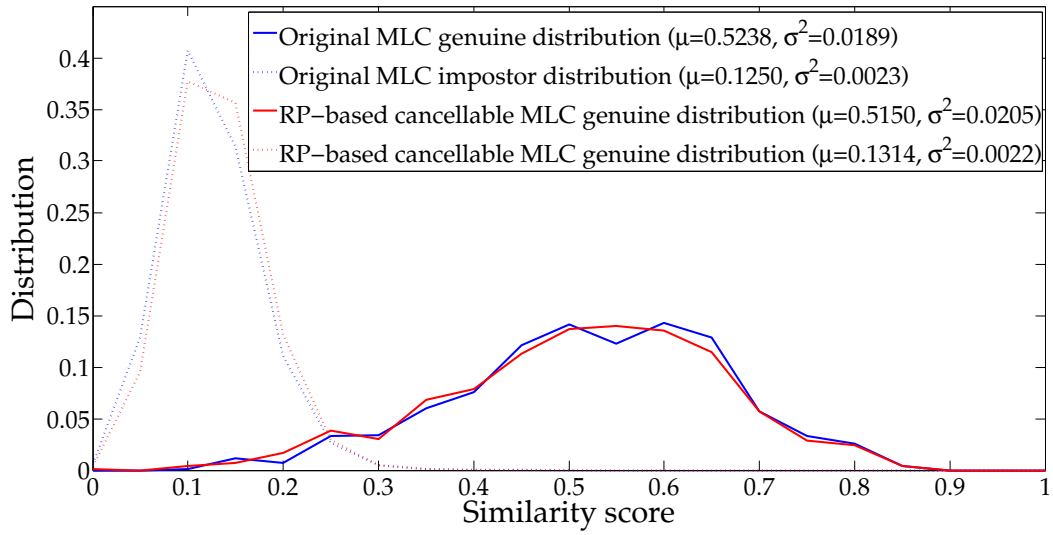
(e) *MLCN*, FVC2004 DB1.

(f) *MLCD*, FVC2004 DB1.

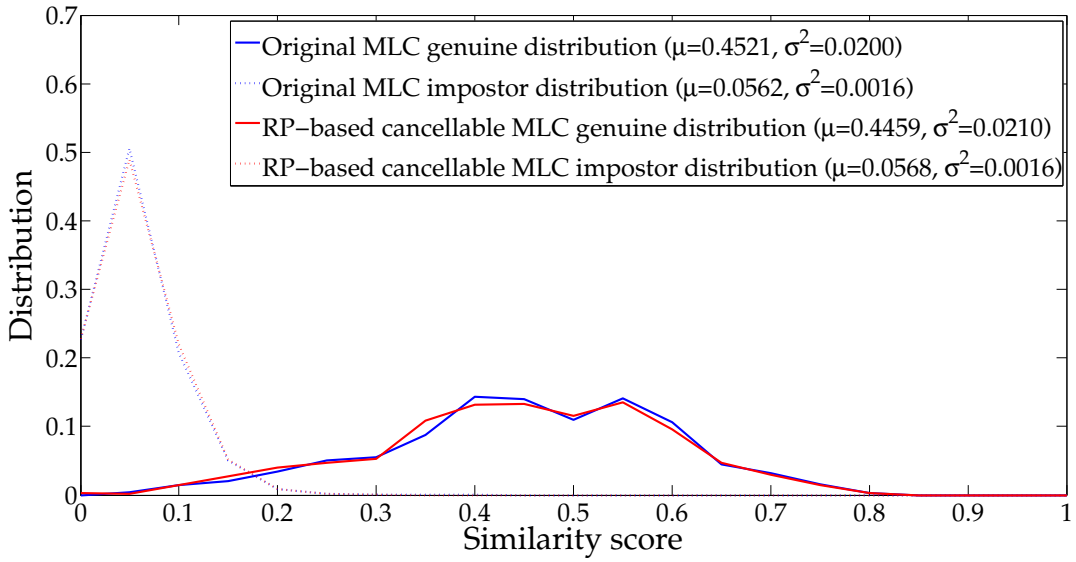
(g) *MLCN*, FVC2004 DB2.

(h) *MLCD*, FVC2004 DB2.

Figure 3.5: EERs of the proposed RP-based revocable MLC under stolen key scenario. The primary axes indicate the EER versus reduced dimension, D_r; while the secondary axes show the EER deterioration ratio and the dimensionality reduction ratio corresponding to the values in the primary axes.



(a) *MLCN*



(b) *MLCD*

Figure 3.6: Stolen key score distribution of RP-based cancellable MLC with $\text{ratio}_{\text{DR}} = 0.4$ for FVC2002 DB1.

has to be understood that comparison of the cancellable fingerprint algorithms using the end results may not be fair as the other modules, such as fingerprint image processing and minutiae extraction, in the recognition system may be different. With this in mind, the benchmarks act as references to evaluate the performance of the proposed method.

Table 3.8: Summary of the recognition accuracy (in terms of EER in %) of the proposed cancellable MLC with $\text{ratio}_{\text{DR}} = 0.4$ compared to other existing cancellable fingerprint methods.

Algorithm	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
Permutation-based cancellable MLC				
MLCN	**2.25**	1.94	8.20	8.05
MLCD	2.32	1.65	**7.54**	**7.82**
RP-based cancellable MLC				
MLCN	3.13	2.44	9.49	8.79
MLCD	2.83	2.25	9.16	8.89
Literature				
Jin et al.[1] [114]	4.36	1.77	24.71	21.82
Jin et al.[2] [176]	3.07	**1.02**	-	-
Zhang et al.[3] [104]	7-8 (FVC2006 DB2)			
Wang and Hu[4] [113]	3.50	4.00	-	-
Lee and Kim[5] [116]	-	-	10.30	9.50
Teoh et al.[6] [112]	2.39		-	-

[1]non-invertible randomized graph-based Hamming embedding.
[2]random projected minutiae vicinity decomposition.
[3]cancellable MCC through combo plate. Results published for FVC2006 DB2 only.
[4]densely infinite-to-one mapping (DITOM) approach.
[5]minutia descriptor by three dimensional spatial quantization.
[6]BioHashing.

3.6.4 Security and Privacy Analyses

In this thesis, the security and privacy of a cancellable fingerprint template is mainly evaluated based on three properties, namely non-invertibility, unlinkability and entropy, each resembling the strength of the proposed MLC algorithm against certain biometric database attack types. All experiments in this section correspond to the best-performing parameters shown in Table 3.4 to 3.7, and with $\text{ratio}_{\text{DR}} = 0.4$ and D_r rounded to the nearest multiple of 50 for RP-based transformation.

Non-invertibility: Resistance against Reverse Attack

If an adversary were to unveil the original fingerprint ISO template from the cancellable template, there are two phases to crack. First, the cancellable template is protected by the cancellable transformation, which is made up of either permutation or RP. Considering that the user-specific key is secure, both transformations are mathematically infeasible to reverse; the adversary can only guess the original MLC by brute force.

However, if the user-specific key is compromised, permutation-based transformation can be easily reversed and the original MLC can be reconstructed exactly as the per-

Table 3.9: Ratio of cardinalities of the proposed cancellable fingerprint template generation scheme, $\frac{\|\boldsymbol{\omega}'\|_0}{\|\hat{\boldsymbol{\omega}}\|_0}$.

Dataset	*MLCN*	*MLCD*
FVC2002 DB1	0.19	0.18
FVC2002 DB2	0.09	0.08
FVC2004 DB1	0.20	0.25
FVC2004 DB2	0.20	0.28

mutation order is known. On the other hand for RP-based transformation, it was shown [177] that if the original signal is sparse (which in the case of MLC, is true), it can be recovered exactly from the transformed template provided that

$$\|\boldsymbol{\omega}'\|_0 \leq \frac{\alpha D_r}{\log D_m}, \tag{3.6.3}$$

where $\|\boldsymbol{\omega}'\|_0$ is the cardinality of a normalized minutia vector and $\alpha > 0$ is some sufficiently small constant. The recovery of discrete Fourier transform (DFT) with randomly selected frequencies is demonstrated by the authors, but the recovery of RP may also be applicable. To recover $\boldsymbol{\omega}'$ from $\hat{\boldsymbol{\omega}}$, one simply needs to solve the convex programming problem:

$$\min \|\boldsymbol{\omega}'\|_1 \quad \text{subject to} \quad \boldsymbol{\omega}'\mathbf{R}_{\perp} = \hat{\boldsymbol{\omega}}. \tag{3.6.4}$$

Since the minutia vectors of one fingerprint are multiplied with the same random matrix, the RP transformation on each minutia vector is considered an independent operation. The equation can be solved by various sparse approximation methods such as matching pursuit [178, 179], basis pursuit [180], the least absolute shrinkage and selection operation (LASSO) approach [181] etc. Further, Candès et al. [177] added that the recovery rate is more than 50% if $\|\boldsymbol{\omega}'\|_0 \leq \|\hat{\boldsymbol{\omega}}\|_0/4$ (or $\frac{\|\boldsymbol{\omega}'\|_0}{\|\hat{\boldsymbol{\omega}}\|_0} \leq 0.25$) and more than 90% if $\|\boldsymbol{\omega}'\|_0 \leq \|\hat{\boldsymbol{\omega}}\|_0/8$ (or $\frac{\|\boldsymbol{\omega}'\|_0}{\|\hat{\boldsymbol{\omega}}\|_0} \leq 0.13$). $\|\hat{\boldsymbol{\omega}}\|_0$ in the ratio is essentially equivalent to D_r as the projection matrix is extracted from a continuous Gaussian

Table 3.9 lists the ratios between the average cardinality values of the minutia vectors (MLC) before and after RP. For most of the cases, the ratio is below 0.25. This indicates that the adversary can recover more than 50% of the unprotected MLCs from the protected template.

Furthermore, the non-invertibility of the proposed cancellable fingerprint partially relies on the MLC algorithm. Assuming that the adversary has knowledge about the MLC algorithm and the parameters, he is able to reverse the minutia vectors into re-

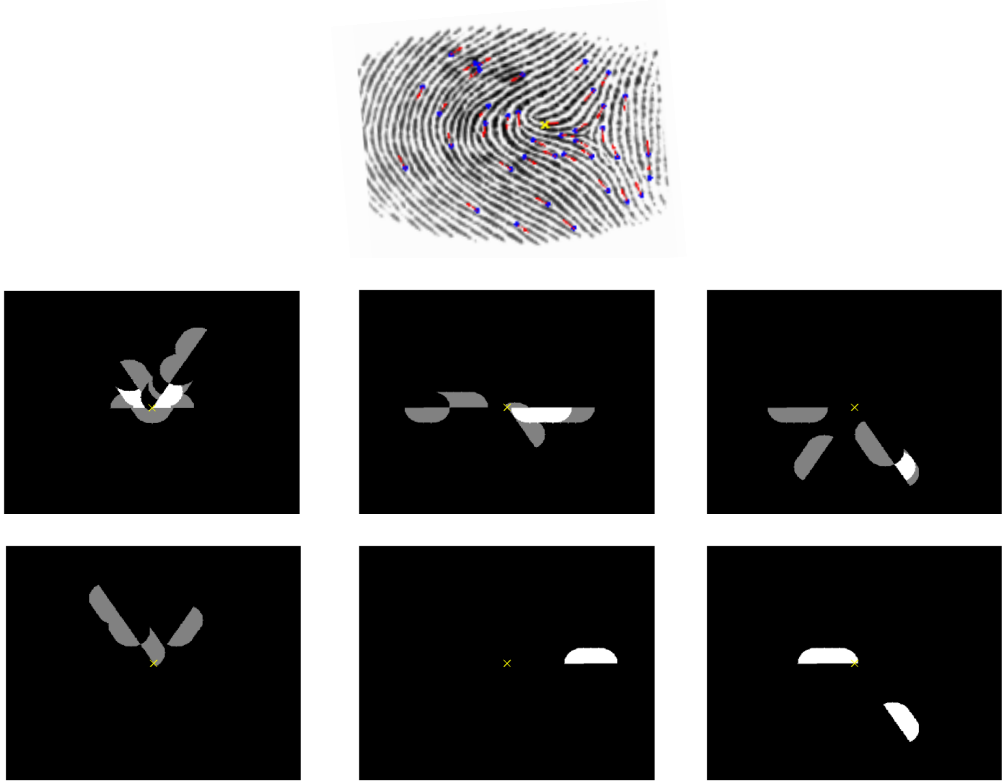


Figure 3.7: An example of reverse attack upon the *MLCN* algorithm. The top row shows the original minutiae set extracted from the fingerprint. The fingerprint is rotated so that the orientation of the reference minutia (marked $\times$) is aligned with 0°. The middle and the bottom rows show the possible locations of the minutiae, divided into six orientation levels ($N_\varphi = 6$), obtained by reversing the MLC of the reference minutia. The pixel intensity of the grayscale images indicates the number of minutiae within the regions, i.e. areas with higher intensity contain more minutiae than areas with lower intensity.

gions where minutiae are possibly located as shown in figure 3.7. Since the minutiae are quantized into multiple levels based on the difference of orientation between the neighbouring minutiae and the reference minutia, the adversary can only obtain the rotated version of the vicinity. The regions obtained for different minutia vectors cannot be inter-related as the orientations of the reference minutiae are independent of each other. Besides, the MLC template is unordered, so the adversary has no clue about the exact location of the recovered regions in the fingerprint. What's more, the vicinity of any one reference minutia contains only part of the entire fingerprint. In conclusion, although the MLC template may reduce the effort of brute force attack on the minutiae set, it is mathematically infeasible to obtain the minutiae set from the template.

Unlinkability: Resistance against Linkage Attack

When an enrolled cancellable template is compromised, it is replaced with a new template associated with a new user-specific key. If the adversary obtains multiple cancellable templates generated from the same fingerprint, he could find the similarity between the templates and learn the pattern of the user's cancellable template. Such attack type is also known as the hill-climbing attack. What's worse, the adversary might be able to gain useful information that assists in reducing the complexity of reverse attack. In this experiment, we intend to measure the separability between templates from the same fingerprint transformed with different keys by following the procedures below [99]:

1. The first sample of every fingerprint is used as the enrolled template. Five versions of the enrolled template is stored, each corresponding to a unique key for cancellable transformation.

2. Each of the remaining seven samples is assigned another ten distinct keys to produce ten unique query templates per sample.

3. Match the query templates against the enrolled templates of the same fingerprint and generate the distribution of matchable minutia vectors between the templates. Recall that two minutia vectors are said to be matchable if the similarity fulfils the condition stated in (3.5.2). The separability between the same-key and the different-key genuine distributions is then computed as:

$$\text{separability} = \frac{|\mu_{\text{DFG}} - \mu_{\text{SKG}}|}{\sqrt{\frac{\sigma^2_{\text{DFG}} + \sigma^2_{\text{SKG}}}{2}}}, \tag{3.6.5}$$

where μ_{DFG} and μ_{SKG}, and σ^2_{DFG} and σ^2_{SKG} are the mean and variance of the same-key and different-key genuine distributions respectively. Note that the same-key genuine distribution is generated the same way as in section 3.6.3, except that number of matchable minutiae is used instead of the matching score.

Table 3.10 shows the separability of MLC and the statistics of the distributions. The separability of MLC is considerably high as compared to the highest separability of 3.2 reported by Lee et al. [99]. Besides, it is noteworthy that both the mean and variance of the different-key distribution are zero's for all algorithms and all datasets. It means that no matchable minutia vectors exist between two cancellable templates generated from

Table 3.10: Separability of the proposed cancellable MLC algorithm expressed in the form of "separability$(\mu_{\text{SKG}}, \sigma^2_{\text{SKG}})[\mu_{\text{DKG}}, \sigma^2_{\text{DKG}}]$".

Algorithm	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
Permutation-based cancellable MLC				
MLCN	3.95 (23.55,71.20) [0,0]	3.96 (30.72,120.52) [0,0]	3.54 (19.87,62.88) [0,0]	3.28 (17.48,56.79) [0,0]
MLCD	3.54 (22.24,78.90) [0,0]	3.28 (27.51,140.59) [0,0]	2.92 (17.76,74.15) [0,0]	2.79 (15.96,65.63) [0,0]
RP-based cancellable MLC				
MLCN	3.85 (23.39,73.77) [0,0]	3.81 (30.07,124.85) [0,0]	3.34 (19.05,65.28) [0,0]	3.26 (17.42,57.05) [0,0]
MLCD	3.47 (22.11,81.28) [0,0]	3.24 (27.25,141.33) [0,0]	2.83 (17.50,76.55) [0,0]	2.80 (15.84,63.77) [0,0]

different keys even if they originate from the same fingerprint. Hence, the proposed cancellable fingerprint is secure against linkage attack.

Entropy: Resistance against Brute Force Attack

It is known that the *MLCN* algorithm produces integer-numbered templates while the *MLCD* algorithm produces real-numbered templates. With the fact that the original MLC template is sparse, the permutation-based cancellable template would also be sparse. We can conveniently consider each element in the template a variable from the discrete space for both permutation-based *MLCN* and *MLCD*. Although *MLCD* is real-numbered, the values are rounded down to the nearest integer (quantization width of 1) for the calculation of the entropy. The entropy of a discrete random variable (also known as Shannon entropy) is defined as:

$$H(\hat{\omega}) = -\sum_{i=1}^{\mathrm{S}(\hat{\omega})} P_{(i)}(\hat{\omega}) \log_2 P_{(i)}(\hat{\omega}) \text{ bits}, \tag{3.6.6}$$

where $\mathrm{S}(\hat{\omega})$ is the support set of $\hat{\omega}$ (an element in the cancellable MLC template $\hat{\mathbf{\Omega}}$) and $P_{(i)}(\hat{\omega})$ is the probability of $\hat{\omega}$ being equivalent to the ith value in the support set.

As for the RP-based cancellable templates, the original MLCs are projected onto a continuous Gaussian space and the entropy of a continuous random variable (also known as differential entropy) is defined as:

$$H(\hat{\omega}) = -\int_{S(\hat{\mathbf{\Omega}})} f(\hat{\omega}) \log_2 f(\hat{\omega}) dx \text{ bits}, \tag{3.6.7}$$

where $f(\hat{\omega})$ is the probability density function (pdf) of the elements in $\hat{\mathbf{\Omega}}$. If the variable observed is taken from a zero-mean Gaussian distribution, (3.6.7) can be rewritten as [182]:

$$H(\hat{\omega}) = \frac{1}{2} \log_2 2\pi e\sigma^2 \text{ bits}, \tag{3.6.8}$$

where e is the Euler's number. Although it is logical to compute the entropy of the RP-based MLC with (3.6.8), it may result in a negative entropy if the variance of the distribution, σ^2 is too small, which is undesirable. Therefore, both (3.6.6) and (3.6.8) are used to compute the entropy of RP-based MLC for comparison in this thesis. The quantization width is set to 0.01 for the calculation of discrete entropy.

As the proposed template is variable-size and unordered, it is impossible to compute the entropy for each element in the template. Instead, we could treat the elements in the template as instances of one random variable and compute the entropy of this variable as an approximation to the average entropy per element.

Table 3.11 presents the entropy of various outcomes of the proposed cancellable fingerprint algorithm. Since the feature length varies for different algorithms and datasets, the average entropy per element is more appropriate for inter-algorithm and inter-dataset comparison. RP-based MLC has significantly higher entropy than permutation-based MLC as the latter is sparse.

For permutation-based algorithms, *MLCD* yields approximately half of the entropy of *MLCN*. While both outputs are sparse, the values of *MLCD* has smaller variance than *MLCN*. This is because two semi-circles enclosing different number of minutiae may result in similar mean distance value from the centre as illustrated in figure 3.8. On the other hand, the entropies for RP-based algorithms are rather similar to each other due to the fact that they are projected onto the Gaussians with identical pdf. The differential entropies are negative as expected because the variance of the output templates is small.

Table 3.11: Entropy (in bits) of the proposed cancellable MLC template. The first number represents the average discrete entropy per element and the second number represents the total discrete entropy of the entire template. The number in bracket is the average differential entropy per element.

Algorithm	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
Permutation-based cancellable MLC				
MLCN	0.43, 390.82	0.37, 543.39	0.45, 668.48	0.42, 612.51
MLCD	0.22, 310.93	0.17, 199.09	0.28, 304.07	0.27, 243.22
RP-based cancellable MLC				
MLCN	2.16, 755.35 (-4.47)	2.03, 1218.02 (-4.95)	2.05, 1230.33 (-4.92)	2.20, 1320.45 (-4.76)
MLCD	2.15, 1290.00 (-4.71)	2.06, 926.16 (-4.59)	2.07, 931.23 (-4.56)	2.33, 815.50 (-4.28)

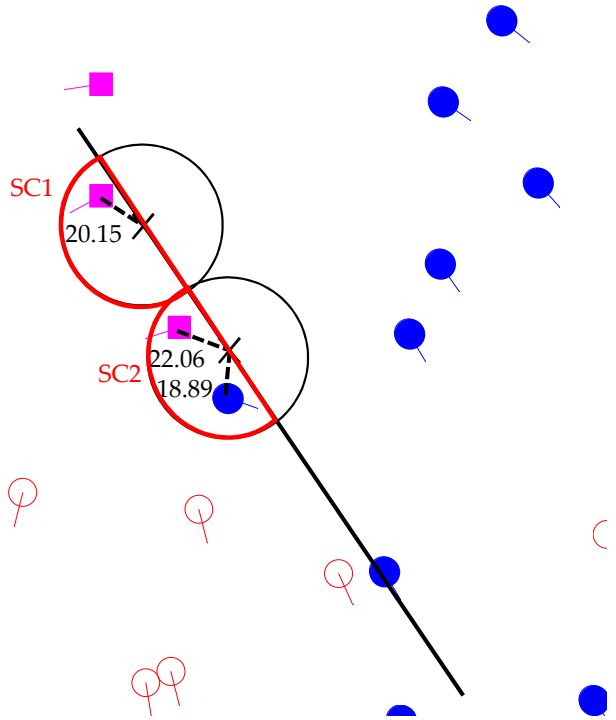


Figure 3.8: The figure shows a portion of the minutiae in a fingerprint. There are two semi-circles labelled with *SC1* and *SC2*, encompassing one minutia and two minutiae respectively. Thus, the *MLCN* code produced are '1' and '2', which are distinctive. On the other hand, the *MLCD* code extracted from the two semi-circles are '20.15' and (22.06+18.89)/2='20.48', which are very close to each other.

3.6.5 Computational Complexity Analysis

In this section, the computational complexity of the MLC algorithm is evaluated in two different measures including time complexity and CPU runtime. MLC can be generated either by recursive search for minutiae within the fixed radius or by performing masking on the minutiae bit map (efficient MLC) as proposed in section 3.3.2.

Nearest neighbour search within fixed radius by recursive sweep involves iterative computations of distance. The number of distance calculations, which comprises of multiplications and additions, depends on the number of minutiae in the fingerprint, N_m. Therefore, the time complexity of producing one component of a minutia vector is $\mathcal{O}(N_m)$. On the other hand, for efficient MLC by masking, the size of the masks and the minutiae map are fixed regardless of any parameter and thus, the number of arithmetic operations is constant, so does the time complexity, i.e. $\mathcal{O}(1)$. In order to generate a complete MLC template for a fingerprint with N_m minutiae and with D_m dimensions per minutia vector, the recursive sweep (or masking operation) needs to be performed $N_m \times D_m$ times. As a result, the time complexity of the normal MLC and efficient algorithms are $\mathcal{O}(N_m^2 D_m)$ and $\mathcal{O}(N_m D_m)$ respectively. Note that although *MLCD* requires additional arithmetic operations for the calculation of mean distance (or an additional distance mask), both *MLCN* and *MLCD* yields the same time complexity. In addition, the permutation-based cancellable transformation has constant time complexity whereas the complexity of RP (matrix multiplication) is bounded by $\mathcal{O}(N_m D_m D_r)$.

Although the bottleneck of the computational complexity from the aspect of time complexity lies with the RP operation. The CPU runtime of permutation and RP are almost identical to each other. Therefore, only RP is used for the following experiment. The RP-based cancellable MLC generation algorithms were performed in the MATLAB environment on Windows 7 with an Intel® Core™ i5-2430M 2.40GHz processor and the recorded CPU runtime is shown in Table 3.12. The best-performing parameters as suggested in section 3.6.2 are applied to each dataset in this experiment. The results echo the time complexity deduced above that the efficient MLC runs much faster than the original MLC. Besides, the difference between *MLCN* and *MLCD* is negligible as the additional computation contributes a very minor part to the overall algorithm. Comparing the results among different datasets, it is evident that the CPU runtime is directly proportional to the length of minutiae vector, D_m as this value varies for each dataset.

The proposed cancellable MLC not only eliminate the computation for fingerprint pre-alignment, with the efficient MLC algorithm, it can be further simplified. Therefore, it consumes less computational power than methods that require fingerprint pre-alignment [116]. Although some alignment-free techniques exist in the literature, repetitive calcu-

Table 3.12: CPU runtime of the MLC generation algorithm with RP transformation.

Algorithm	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
MLC (by recursive sweep)				
MLCN	6.36s	9.79s	8.21s	6.11s
MLCD	6.68s	9.85s	8.80s	6.44s
Efficient MLC (by masking)				
MLCN	2.94s	3.30s	3.33s	2.79s
MLCD	2.99s	3.31s	3.54s	2.90s

lation of some local attributes for example the lengths and angles of triplets [113, 114, 176] is also resource-consuming.

3.7 Summary

In this chapter, a novel cancellable fingerprint generation method has been proposed, coined as the MLC. MLC is rotation- and translation-invariant as the positions of the minutiae are relative to the reference minutia. In addition, MLC is incorporated with the idea of fixed-radius descriptor and is robust against local non-linear distortions. The main concept of MLC generation branches out into the integer-numbered version which record the number of minutiae within the fixed-radius semi-circles (*MLCN*) and the real-numbered version which takes the mean of distances from all minutiae within the radius to the centre as the feature code (*MLCD*). Besides, two distinct cancellable transformations have been used to realize the cancellability of the MLC template, namely permutation and RP.

Experimental results show that both algorithms are able to achieve 0% EER for genuine-key scenario. This is supported by the none-overlapping between the genuine distribution and the impostor distribution as illustrated in figure 3.4. According to 3.8, *MLCD* performs better than *MLCN* for most datasets, proving that the mean distance value is a better feature in describing the semi-circle neighbourhoods created for MLC generation compared to minutiae count. Moreover, permutation-based cancellable MLC is able to maintain the performance of the original MLC while RP-based templates suffer from EER deterioration. This is caused by the dimensionality reduction effect of RP, where the features are restricted in a much lower dimension. Fortunately, the deterioration can be kept minimum at an appropriate dimension.

Furthermore, comprehensive studies have been done regarding the security and privacy issues of the proposed scheme. Results show that it excels in both unlinkability and entropy which prove its resistance against linkage attack and brute force attack respectively. Although the compromise of the algorithm parameters may reduce the effort of reversing the MLC template by brute force, the MLC algorithm remain mathematically irreversible. Hence, the proposed algorithm has moderate non-invertibility. Comparing the two MLC types, *MLCN* and *MLCD* yield almost similar security and privacy strength.

From the aspect of computational complexity, the proposed efficient MLC algorithm has successfully reduced the time complexity from $\mathcal{O}(N_m^2 D_m)$ to $\mathcal{O}(N_m D_m)$, or the running time by more than half. Both *MLCN* and *MLCD* share the same time complexity and have similar running time.

In a nutshell, the proposed cancellable fingerprint template generation scheme fulfils the criteria set in 1.3. Not only that the recognition accuracy is comparable to existing methods in the literature, the proposed scheme also provides high security and privacy, and is computationally efficient. The *MLCD* algorithm can be seen as an offer of trade-off between performance and security.

CHAPTER 4

Minutiae Set to Feature Vector (S2V) Transformation via Kernel Subspace Analysis

4.1 Background

A minutiae-based cancellable fingerprint template, coined as the MLC has been introduced in the previous chapter. Like most minutiae-based methods discussed in section 2.2.1, MLC produces an unordered and variable size set of minutia vectors. Such biometric template hinders its adaptability in several applications such as bio-cryptosystems, continuous classification in fingerprint indexing[1] [185], vector component-specific analysis for dynamic quantization and SVM classifier[2] and thus, S2V transformation is essential.

In this chapter, a novel S2V transformation technique based on kernel subspace analysis is proposed. Kernel subspace analysis turns linear subspace models into non-linear models by applying the kernel trick. With the use of kernel functions, kernel trick operates in a high-dimensional and implicit feature space without actually computing the values of the data in that space. First of all, the two possible choices of linear subspace

[1] Indexing of minutiae-based fingerprint templates (e.g. [183,184]) usually involves repetitive matching of local fingerprint structures. More efficient fingerprint indexing method employ hash-based technique [184] to speed up the fingerprint retrieval process. If the fingerprint template is globally ordered and fixed-length, it can be represented as a point in the continuous space and direct comparison between two templates using standard distance metrics is made possible.

[2] A SVM model represents the input data as points in a *fixed-dimension* space, which are then mapped into another *fixed-dimension* space so that the samples of separate classes are well-divided.

analysis methods commonly used in the realm of biometrics are briefly described below:

- Principal components analysis (PCA): PCA defines an orthogonal subspace that optimally describes the variance among the input data. It can be done by performing eigenvalue decomposition on the data covariance matrix. The final outcome is computed by projecting the input data onto the mutually orthogonal eigenvectors, sorted in descending order of the corresponding eigenvalues. This essentially puts the greatest variance in the first dimension of the projected output so that it can account for as much variability in the input data as possible. This is particularly useful in dimensionality reduction where only a small number of dimensions are required to preserve the information in the original data. Besides the standard linear PCA, other variants of the PCA model include nonlinear PCA [186], kernel PCA [187] and sparse PCA [188]. PCA and its variants have been successfully implemented for face recognition [189–191] and palmprint recognition [192] for feature extraction.

- Linear discriminant analysis (LDA): While PCA does not take into account the classes of the input data, LDA aims at deriving linear combinations of variables which best describe the classes of data. The covariance matrices of different classes are calculated separately and pooled prior transformation through the discriminant function. LDA is also known as the Fisher's linear discriminant. Fisherfaces [193] and kernel-based Fisherfaces [194] are two instances of the application of LDA model in biometric authentication.

The major difference between LDA and PCA is that the former requires class-labelling of the training observations. In the case of fingerprint recognition, a class represents a unique fingerprint. In practice, the number of fingerprints (or the number of users) in the system database may grow when new users are enrolled into the system. This makes the training process of LDA intractable as new classes would have to be added. Therefore among the statistical analysis methods, the PCA model is more suitable for the S2V transformation purpose. However, linear PCA does not accept unordered and variable-size dataset as the input. In this thesis, we exploit the flexibility of kernel PCA (KPCA) in its input data type to achieve the transformation. Part of the work presented in this chapter has been published [195].

4.2 Related Work on S2V Transformation

Sutcu et al. [107] used a local point aggregation approach which constructs random cuboids on the three-dimensional space (x-coordinate, y-coordinate and orientation) and produce a histogram based on the number of minutiae enclosed in each cuboid. This work was extended by Nagar et al. [108], in which more discriminative features were used for binning, such as the distance from minutiae to the nearest boundary, the average and standard deviation of minutiae coordinates. Besides, highly-correlated bits are eliminated to improve the performance. Moreover, Gudkov and Ushmaev [196] employed random minutiae cluster selection and performed matching on the fixed number of selected minutiae only. However, this method suffers great loss in accuracy due to the possibility of indistinctive minutiae being chosen. These histogram-based methods operate on the spatial domain of minutiae, which directly utilizes the ISO template of minutiae in S2V transformation, and thus require fingerprint pre-alignment.

Moreover, Bringer and Despiegel [47] proposed a method to generate binary fingerprint feature vector from minutiae vicinities, in which each minutia is represented by a vicinity vector. These vicinities are then matched against a set of N representative vicinities. A N-bit binary string is acquired by thresholding the maximum score for each representative vicinity. Another histogram-based approach was introduced by Farooq et al. [100] to convert minutiae set into a fixed-length binary template. Initially, the attributes of each minutiae triplet are converted into a N-bit string. The final bit-string is generated by identifying the triplet patterns which only appear once in the fingerprint out of the histogram of length 2^N. A more sophisticated technique, namely the k-means algorithm was used by Vij and Namboodiri [197] to determine the representative features in order to attain better performance.

Luo et al. [198] extends the idea of MCC local descriptor into global MCC. Instead of encoding the vicinity of each minutia, the global MCC encodes the entire fingerprint region based on reference points so that only a fixed number of global MCCs are produced. The reference points should be stable across different samples of a fingerprint.

Further, Xu et al. [136] presented a novel fixed-length fingerprint feature known as the spectral minutiae. In their method, each minutia is expressed by an impulse function and the spectral representation of the minutiae is obtained by performing discrete Fourier transform on the aggregated impulses in the polar-logarithmic domain. Hence, the feature vector generated is invariant to translation, rotation and scaling. Instead of

magnitude spectrum, Nandakumar [137] used the minutiae phase spectrum representation to transform an unordered minutiae set into a fixed-length vector, resulting in a better accuracy. However, the improvement in accuracy is marginal without pre-alignment using the local features. These local features are represented in a variable-size set.

In summary, the S2V transformation methods in the literature can be primarily categorized into: *i*) the histogram-based approach [47, 100, 107, 108, 196–198] which perform minutiae binning based on pre-defined prototypes; and *ii*) the spectral analysis approach [136, 137] which converts the minutiae map into a bit-plane and obtain the frequency spectrum via Fourier transform. For the former approach, the bin definition algorithm used for histogram generation is vital in determining the information preservation property of the S2V transformation and thus, has to be prudently designed. As for the latter, the minutiae are originally represented in the continuous space. There are bound to be errors when discrete Fourier transform is performed on a continuous image.

Apart from fingerprint recognition, S2V conversion is also much researched in other fields of study. A multi-interface tool named Sally [199] was developed to embed strings in vector spaces by using the bag-of-words model. The tool demonstrated high efficiency even with large-scale data, but the classification performance was not discussed. Spillmann et al. [200] proposed a strings to real vector transformation by calculating the edit distances of the query string to the predefined prototypes. The prototype selection strategy was designed so that there is no redundancies and outliers, and that the prototypes are uniformly distributed. The transformed data could achieve 95% recognition rate in bold-face digit recognition by using SVM with radial basis function (RBF) kernel. Furthermore, Sonnenburg et al. [201] demonstrated a kernel-based feature vector extraction method on DNA sequences.

4.3 Preliminary: KPCA

The technique of kernel substitution is a way of observing an arbitrary mapping from the data space, $\{\mathbf{X}_{(i)}\}$ ($\mathbf{X}_{(i)} \in \mathbb{R}^{D_x}$) into the feature space, $\{\Phi(\mathbf{X}_{(i)})\}$ ($\Phi(\mathbf{X}_{(i)}) \in \mathbb{R}^{D'_x}$) without having to compute the mapping explicitly, with $i \in [1, N]$ and N being the number of data observations in general. Combining the kernel trick with PCA obtains a non-linear generalization of PCA called KPCA [187]. In the transformed feature space,

the covariance matrix is

$$\mathbf{Cov} = \frac{1}{N}\sum_{i=1}^{N}\Phi(\mathbf{X}_{(i)})\Phi(\mathbf{X}_{(i)})^{\mathrm{T}} \tag{4.3.1}$$

for $\mathbf{Cov} \in \mathbb{R}^{D'_x \times D'_x}$ and thus the eigenvalue equation is

$$\boldsymbol{\lambda\varepsilon} = \mathbf{Cov}\boldsymbol{\varepsilon}, \tag{4.3.2}$$

where $\boldsymbol{\varepsilon}$ and $\boldsymbol{\lambda}$ are the eigenvectors and eigenvalues of $\Phi(\mathbf{X}_{(i)})$. $\boldsymbol{\varepsilon}$ is given by a linear combination of $\Phi(\mathbf{X}_{(i)})$ and it can be written in the form

$$\boldsymbol{\varepsilon} = \sum_{i=1}^{N}\hat{\boldsymbol{\varepsilon}}\Phi(\mathbf{X}_{(i)}). \tag{4.3.3}$$

By substituting (4.3.1) and (4.3.3) into (4.3.2) and multiplying both sides by $\Phi(\mathbf{X}_{(i)})^{\mathrm{T}}$ we get

$$N\boldsymbol{\lambda}\hat{\boldsymbol{\varepsilon}} = \mathbf{K}\hat{\boldsymbol{\varepsilon}}, \tag{4.3.4}$$

where $\mathbf{K} \in \mathbb{R}^{N\times N}$ is the Gram matrix and $\hat{\boldsymbol{\varepsilon}}$ is the normalized eigenvectors of $\mathbf{K}$. The Gram matrix or the kernel function can be expressed in the inner product form

$$\mathbf{K} = k(\mathbf{X},\mathbf{Y}) = \langle\Phi(\mathbf{X}),\Phi(\mathbf{Y})\rangle = \Phi(\mathbf{X})^{\mathrm{T}}\Phi(\mathbf{Y}). \tag{4.3.5}$$

KPCA allows one to calculate the projection of the feature vector onto the principal components $\boldsymbol{\varepsilon}_p \subset \boldsymbol{\varepsilon}$ through the kernel function

$$\mathbf{Y} = \boldsymbol{\varepsilon}_p^{\mathrm{T}}\Phi(\mathbf{X}) = \hat{\boldsymbol{\varepsilon}}_p k(\mathbf{X},\mathbf{Y}), \tag{4.3.6}$$

where $\mathbf{Y} \in \mathbb{R}^{N_p}$ and $\hat{\boldsymbol{\varepsilon}}_p \subset \hat{\boldsymbol{\varepsilon}}$ with N_p being the number of principal components extracted. The polynomial kernel $k(\mathbf{X},\mathbf{Y}) = (\mathbf{X}^{\mathrm{T}}\mathbf{Y})^n$ [187] and the Gaussian kernel $k(\mathbf{X},\mathbf{Y}) = \exp(-\|\mathbf{X}-\mathbf{Y}\|^2/2\sigma^2)$ [202] has been successfully applied in KPCA.

KPCA has been widely used in various pattern classification problems, such as novelty detection [203], active shape models detection [204], fault detection of non-linear processes [205] and face recognition [190]. In all the examples, KPCA is used to extract features from the original fixed-dimension data space. In this thesis, however, we employ KPCA for an unconventional purpose — to convert variable-size data into fixed-length representation.

4.4 Nomenclature

Symbol	Description
$\mathbf{\Omega}$, $\hat{\mathbf{\Omega}}$	MLC fingerprint template before and after cancellable transformation
N_{spu}	number of training samples per user
N_u	number of users
N_t	total number of training samples
$\mathbf{\Omega}_{\text{train}}$	training MLC templates for KPCA
$\mathbf{K}_{\text{train}}$	training kernel matrix
S_g	global similarity between two fingerprints
σ	scaling parameter of Gaussian kernel
$\hat{\boldsymbol{\varepsilon}}$	eigenvectors of the training kernel matrix
$\hat{\boldsymbol{\varepsilon}}_p \in \mathbb{R}^{N_t \times N_p}$	selected principal components of the training kernel matrix
N_p	number of principal components selected, also the dimension of the fingerprint template after KPCA
$\mathbf{K}_{\text{test}}$	testing kernel matrix
$\mathbf{V}_{\text{KPCA}} \in \mathbb{R}^{N_p}$	fixed-length fingerprint template generated through KPCA
$\hat{\mathbf{V}}_{\text{KPCA}} \in \mathbb{R}^{D_r}$	cancellable fixed-length fingerprint template generated through KPCA
D_r	dimension of the cancellable fixed-length template

4.5 Proposed S2V Transformation

Regardless of the original data structure, KPCA is able to produce a fixed-length feature by performing PCA on a transformed data space without computing the transformation explicitly. Here, we exploit this characteristic of KPCA to convert the MLC template into a fixed-length feature vector. It may apply to both cancellable ($\hat{\mathbf{\Omega}}$) and non-cancellable MLC ($\mathbf{\Omega}$). The non-cancellable template is used for methodology illustration in this section. As conventional kernel functions (i.e. polynomial kernel, radial

basis functions and sigmoid kernel) cannot be directly implemented on variable-size data, we introduce a novel non-linear kernel function specifically for minutiae-based algorithms.

Figure 4.1 summarizes the operations involved in the proposed KPCA-based S2V transformation. It consists of a training stage and a transformation stage as described below:

1. Training stage: The pseudo code of the training stage is given in Algorithm 4.1. Suppose N_{spu} samples are taken from every user, the total number of training samples is $N_t = N_{spu}N_u$, where N_u is the number of users. The Gram matrix $\mathbf{K}_{\text{train}} \in \mathbb{R}^{N_t \times N_t}$ is first computed by using the Gaussian-like matching function of MLC

$$\begin{aligned} K_{\text{train}(ij)} &= k(\mathbf{\Omega}_{\text{train}(i)}, \mathbf{\Omega}_{\text{train}(j)}) \\ &= \exp\{-[1 - S_g(\mathbf{\Omega}_{\text{train}(i)}, \mathbf{\Omega}_{\text{train}(j)})]^2/2\sigma^2\}, \end{aligned} \tag{4.5.1}$$

   where $\mathbf{\Omega}_{\text{train}(i)}$ refers to a training sample from the training set, $\mathbf{\Omega}_{\text{train}}$ and S_g is the matching score between two variable-size templates as derived in (3.5.3). With $\mathbf{K}_{\text{train}}$ obtained, the eigenvectors $\hat{\boldsymbol{\varepsilon}}$, and subsequently the principal components $\hat{\boldsymbol{\varepsilon}}_p$ can be extracted using (4.3.4), where $\hat{\boldsymbol{\varepsilon}}_p \in \mathbb{R}^{N_t \times N_p}$ with N_p denoting the desired number of principal components and $N_p < N_t$. Since the minutiae extracted for a fingerprint are different each time the fingerprint is scanned, the term $S_g(\mathbf{\Omega}_{\text{train}(i)}, \mathbf{\Omega}_{\text{train}(j)})$ in (4.5.1) produces non-binary similarity score ranging from 0 to 1, resulting in a continuous Gram matrix rather than a discrete matrix. Also, note that large N_t results in large matrix $\mathbf{K}_{\text{train}}$ and may cause memory shortage error when performing eigen decomposition. In such case, incremental eigen decomposition [206] or incremental PCA [207] algorithms may be applied. Some of these algorithms claim to have achieved comparable performance as the original PCA.

2. Transformation stage: Algorithm 4.2 depicts the pseudo algorithm of the transformation stage. Given an input fingerprint template, $\mathbf{\Omega}$, the corresponding kernel matrix $\mathbf{K}_{\text{test}}$ is constructed so that the matrix elements are computed as follow:

$$\begin{aligned} K_{\text{test}(i)} &= k(\mathbf{\Omega}_{\text{train}(i)}, \mathbf{\Omega}) \\ &= \exp\{-[1 - S_g(\mathbf{\Omega}_{\text{train}(i)}, \mathbf{\Omega})]^2/2\sigma^2\}, \end{aligned} \tag{4.5.2}$$

   for $i \in [1, N_t]$ and thus, $\mathbf{K}_{\text{test}} \in \mathbb{R}^{N_t}$. After that, substitute $\hat{\boldsymbol{\varepsilon}}_p$ and $\mathbf{K}_{\text{test}}$ into (4.3.6) to obtain a fixed-length feature vector, hereafter denoted as $\mathbf{V}_{\text{KPCA}} \in \mathbb{R}^{N_p}$.

Algorithm 4.1: KPCA-based S2V training

Data: $\mathbf{\Omega}_{\text{train}}, N_p, \sigma$
Result: $\hat{\boldsymbol{\varepsilon}}_p$

begin
  $N_t \longleftarrow \text{size}(\mathbf{\Omega}_{\text{train}})$
  $\mathbf{K}_{\text{train}}(N_t, N_t) \longleftarrow 1$
  for $i \in [2, N_t]$ **do**
    for $j \in [1, i-1]$ **do**
      $\mathbf{K}_{\text{train}}[i,j] \longleftarrow \text{MLCMatch}(\mathbf{\Omega}_{\text{train}}[i], \mathbf{\Omega}_{\text{train}})[j]$ `// refer to Algorithm 3.1 for details of MLCMatch()`
      $\mathbf{K}_{\text{train}}[i,j] \longleftarrow \exp(-(1-\mathbf{K}_{\text{train}}[i,j])^2/2\sigma^2)$
    end
  end
  $\mathbf{K}_{\text{train}} \longleftrightarrow \mathbf{K}_{\text{train}} + \mathbf{triu}(\mathbf{K}_{\text{train}})^{\text{T}}$ `// make` $\mathbf{K}_{\texttt{train}}$ `symmetric by reflecting the upper triangle along the diagonal`
  $\hat{\boldsymbol{\varepsilon}}_p \longleftrightarrow \mathbf{eigvector}(\mathbf{K}_{\text{train}})[1 : N_p]$
end

Algorithm 4.2: KPCA-based S2V transformation

Data: $\mathbf{\Omega}, \mathbf{\Omega}_{\text{train}}, \hat{\boldsymbol{\varepsilon}}_p, \sigma$
Result: $\mathbf{V}_{\text{KPCA}}$

begin
  $N_t \longleftarrow \text{size}(\mathbf{\Omega}_{\text{train}})$
  $\mathbf{K}_{\text{test}} \longleftarrow 0$
  for $i \in [1, N_t]$ **do**
    $\mathbf{K}_{\text{test}}[i] \longleftarrow \text{MLCMatch}(\mathbf{\Omega}_{\text{train}}[i], \mathbf{\Omega})$ $\mathbf{K}_{\text{test}}[i] \longleftarrow \exp(-(1-\mathbf{K}_{\text{test}}[i])^2/2\sigma^2)$
  end
  $\mathbf{V}_{\text{KPCA}} \longleftarrow \mathbf{K}_{\text{test}} \times \hat{\boldsymbol{\varepsilon}}_p$
end

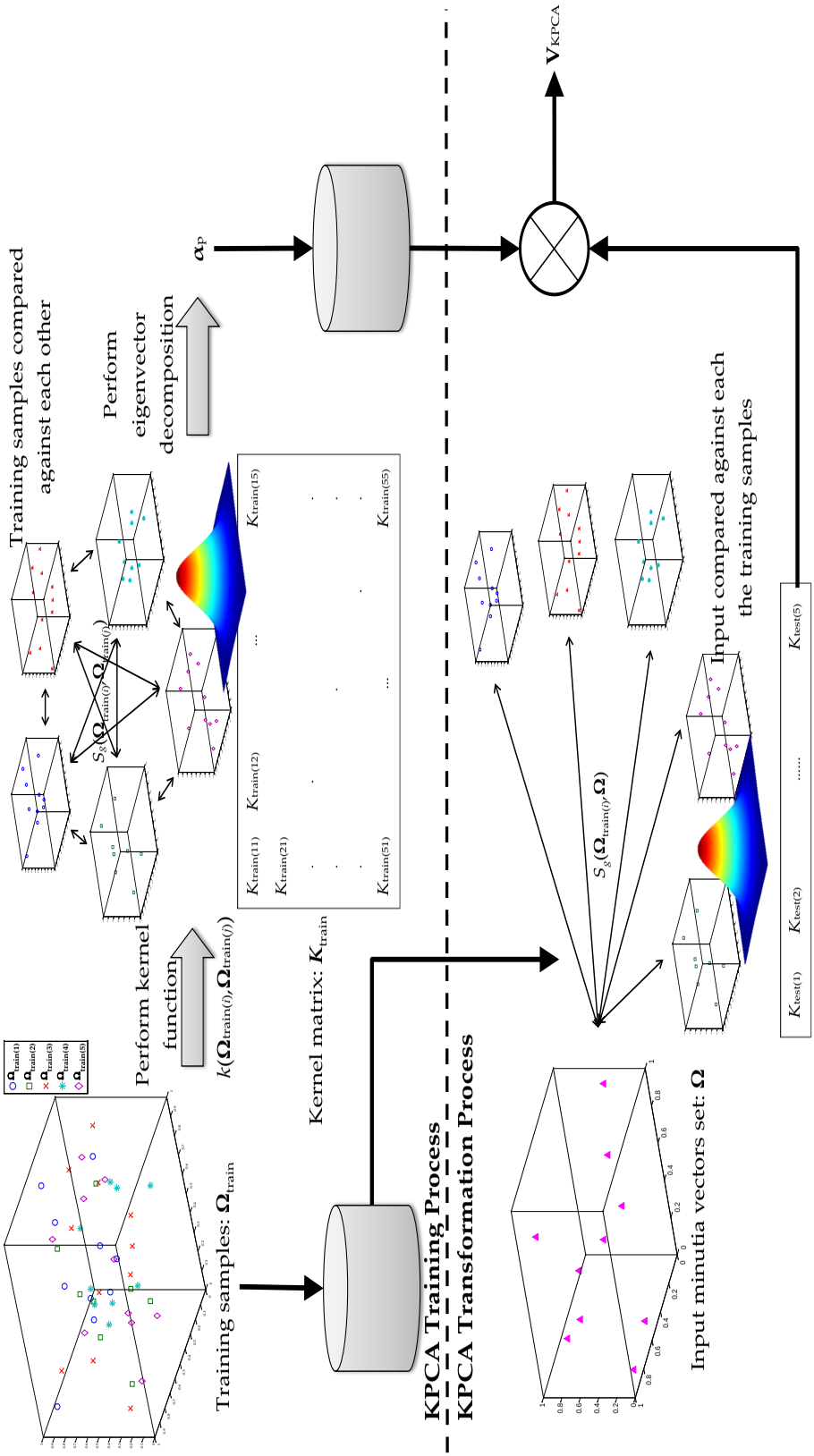


Figure 4.1: An illustration of the KPCA-based S2V transformation. In the example, five training samples, each with different number of minutiae, are used for kernel construction and principal components extraction. In the transformation process, the final product, $\mathbf{V}_{KPCA}$ is a fixed-length vector regardless of the number of minutiae in the input fingerprint.

Since the kernel function used is not a conventional kernel function, it is necessary to verify that the resulting training kernel matrix, $\mathbf{K}_{\text{train}}$ is symmetric and positive semi-definite. We examine these criterion through Mercer's condition [208] as elaborated in Theorem 1 (appendix B) even though it is said that non-Mercer kernels do not necessarily deteriorate the performance [209].

In addition to Lemma 1 (appendix B), we propose the following to validate the kernel function used:

Proposition 1. *Let $k_1(x,y)$ be an admissible kernel over the data space $X \times X$, $X \subseteq \mathbb{R}^n$ and $0 \leq k_1(x,y) \leq 1$, then $k(x,y) = \exp(-k_1(x,y))$ is also a kernel.*

Proof for Proposition 1. By substituting the proposed kernel into the left hand side of Eq. (B.0.2), with Taylor series expansion we get

$$\begin{aligned} \iint k(x,y)g(x)g(y)dxdy &= \iint \exp(-k_1(x,y))g(x)g(y)dxdy \\ &= \iint \sum_{i=0}^{\infty} \frac{(-k_1(x,y))^i}{i!} g(x)g(y)dxdy \\ &= \iint \sum_{j=0}^{\infty} \frac{(k_1(x,y))^{2j}}{(2j)!} g(x)g(y)dxdy \\ &\quad - \iint \sum_{j=0}^{\infty} \frac{(k_1(x,y))^{2j+1}}{(2j+1)!} g(x)g(y)dxdy. \end{aligned} \tag{4.5.3}$$

Since $k_1(x,y)$ is a kernel and by using Lemma 1, we can deduce that $\int\int \frac{(k_1(x,y))^i}{i!} g(x)g(y)dxdy \geq 0$ for $i \in \mathbb{N}$. Also, since $0 \leq k_1(x,y) \leq 1$, we have

$$\iint \frac{(k_1(x,y))^{2j}}{(2j)!} g(x)g(y)dxdy \geq \iint \frac{(k_1(x,y))^{2j+1}}{(2j+1)!} g(x)g(y)dxdy, \tag{4.5.4}$$

and hence

$$\iint \exp(-k_1(x,y))g(x)g(y)dxdy \geq 0 \tag{4.5.5}$$

□

Consequently, the proof showing that the kernel function in Eq. (4.5.1) is a valid kernel is as follow:

Proof that Eq. (4.5.1) *is a valid kernel.* According to the steps of matching algorithm discussed in section 3.5, we can decompose Eq. (4.5.1) into

$$\begin{aligned} K_{\text{train}(ij)} &= k(\mathbf{\Omega}_{\text{train}(i)}, \mathbf{\Omega}_{\text{train}(j)}) \\ &= \exp\{-[1 - S_g(\mathbf{\Omega}_{\text{train}(i)}, \mathbf{\Omega}_{\text{train}(j)})]^2 / 2\sigma^2\} \\ &= \exp\left[-\left(1 - \frac{\sum_{i,j} <\boldsymbol{\omega}_{\text{train}(im)}, \boldsymbol{\omega}_{\text{train}(jn)}> / (\|\boldsymbol{\omega}_{\text{train}(im)}\|_2^2 + \|\boldsymbol{\omega}_{\text{train}(jn)}\|_2^2)}{\min(N_{m(i)}, N_{m(j)})}\right)^2 / 2\sigma^2\right] \\ &= \exp(-1/2\sigma^2) \exp\left[-\left(\frac{\sum_{i,j} <\boldsymbol{\omega}_{\text{train}(im)}, \boldsymbol{\omega}_{\text{train}(jn)}> / (\|\boldsymbol{\omega}_{\text{train}(im)}\|_2^2 + \|\boldsymbol{\omega}_{\text{train}(jn)}\|_2^2)}{\sqrt{2}\sigma \min(N_{m(i)}, N_{m(j)})}\right)^2\right] \\ &\quad \exp\left(\frac{\sum_{i,j} <\boldsymbol{\omega}_{\text{train}(im)}, \boldsymbol{\omega}_{\text{train}(jn)}> / (\|\boldsymbol{\omega}_{\text{train}(im)}\|_2^2 + \|\boldsymbol{\omega}_{\text{train}(jn)}\|_2^2)}{\sigma^2 \min(N_{m(i)}, N_{m(j)})}\right), \end{aligned} \tag{4.5.6}$$

where m and n represents the indices of any matchable minutiae pair in $\mathbf{\Omega}_{\text{train}(i)}$ and $\mathbf{\Omega}_{\text{train}j}$ respectively and $N_{m(i)}$ and $N_{m(j)}$ are the numbers of minutiae in the two fingerprint templates. Plus, the sum of squared ℓ^2 norm, $(\|\boldsymbol{\omega}_{\text{train}(im)}\|_2^2 + \|\boldsymbol{\omega}_{\text{train}(jn)}\|_2^2)$ is always positive. As the inner product in (4.5.6) is already a known kernel, by using Lemma 1 and Proposition 1, it is plain that $k(\mathbf{\Omega}_{\text{train}(i)}, \mathbf{\Omega}_{\text{train}(j)})$ is a valid kernel. □

Now that the proposed kernel function is validated, it is also important to make sure that the value of σ in (4.5.1) and (4.5.2) is not too small or too large to avoid ill-formed kernel matrix. As $S_g(\mathbf{\Omega}_{\text{train}(i)}, \mathbf{\Omega}_{\text{train}(j)})$ (or $S_g(\mathbf{\Omega}_{\text{train}(i)}, \mathbf{\Omega})$) is a normalized similarity score between two positive vectors, the value of $1 - S_g(\mathbf{\Omega}_{\text{train}(i)}, \mathbf{\Omega}_{\text{train}(j)})$ (or $1 - S_g(\mathbf{\Omega}_{\text{train}(i)}, \mathbf{\Omega})$) ranges from 0 to 1. If σ is too small or too large, it becomes the dominant factor in the exponential and equalizes the output value regardless of the similarity score. Besides, a good kernel with the intuition of similarity measure should satisfy the rule of thumb [210]:

$$(k(x,y)|(x=y)) > (k(x,y)|(x \neq y)). \tag{4.5.7}$$

4.6 Experiments and Analyses

4.6.1 Testing Protocol

Note that the same fingerprint datasets (FVC2002 DB1, FVC2002 DB2, FVC2004 DB1 and FVC2004 DB2 as mentioned in section 3.6.1) are used throughout this thesis for fair comparison. In Chapter 3, two MLC algorithms have been proposed, namely *MLCN*

and *MLCD*, and the latter has been proven to be more superior than the former in performance with insignificant loss in the security. Therefore, the *MLCD* algorithm, along with its best-performing parameters set as concluded in Chapter 3, is used in the experiments in this chapter.

Also, cancellable transformation by RP provides better security than by permutation with minimal performance trade-off, hence RP is chosen as the revocation method (refer to section 3.4 for the details of RP). The experiments in this chapter follow the phase-wise sequence highlighted in Figure 4.2. Instead of directly appending the S2V transformation to the cancellable template generation procedure proposed in Chapter 3, it is computationally more efficient to perform cancellable transformation after S2V conversion.

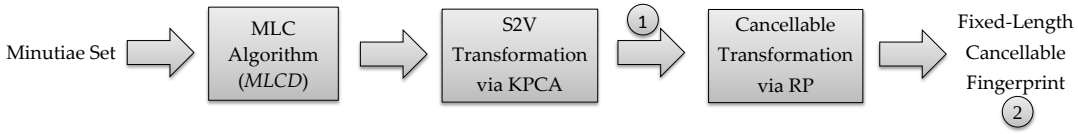


Figure 4.2: EERs of the proposed fixed-length representation of MLC via KPCA-based S2V transformation while altering N_p and σ. The examinable outcomes are labelled with *1* and *2*.

The proposed S2V transformation via KPCA is a training-based algorithm. Out of the 8 samples per fingerprint, 6[3] are allocated for training purpose while the remaining samples are used for matching. That makes up 600 training samples and 200 testing samples in each dataset. As a result, there are 100 genuine matchings and 9900 impostor matchings for each dataset. The testing procedure is iterated fivefold with different randomly selected training and matching samples to obtain results that are more representative and unbiased. Also note that since KPCA operates in the inner product space, the fixed-length vectors are compared with the inner product similarity. On the other hand, a more common ℓ^2-based metric, namely one minus normalized Euclidean distance, is used to match two cancellable templates:

$$S = 1 - \frac{\|\hat{\mathbf{V}}'_{KPCA} - \hat{\mathbf{V}}''_{KPCA}\|_2}{\|\hat{\mathbf{V}}'_{KPCA}\|_2 + \|\hat{\mathbf{V}}''_{KPCA}\|_2}, \tag{4.6.1}$$

where $\hat{\mathbf{V}}'_{BoM}$ and $\hat{\mathbf{V}}''_{BoM}$ are two instances of the proposed KPCA-based cancellable fingerprint representation.

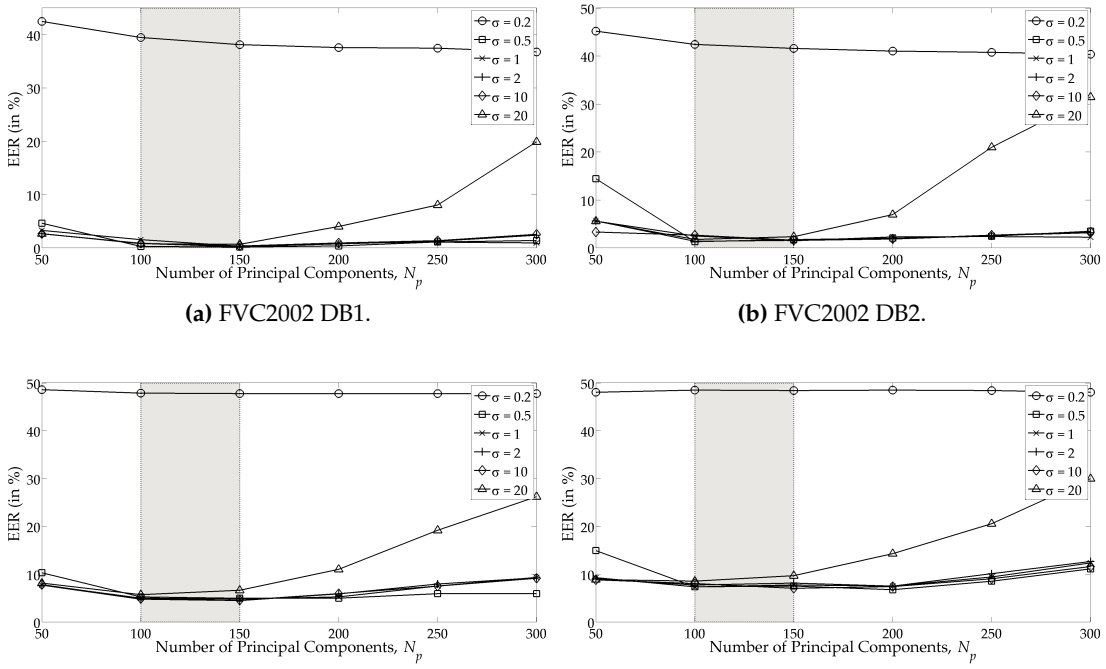
[3]Preliminary experiments had been done to determine the number of training samples required to achieve optimal performance with the fingerprint datasets available.

4.6.2 Effect of KPCA Parameters on Performance

In this section, the effect of changing the number of selected principal components, N_p and the Gaussian parameter, σ on the performance of KPCA-based S2V transformation is studied. The results discussed in this section correspond to the first examinable outcome highlighted in Figure 4.2. Figure 4.3 depicts the performance of the fixed-length representation on different datasets with N_p ranging from 50 to 300 with a step size of 50 and five σ values from 0.2 to 20. The performance suffers a drastic fall when the value of σ is extremely small ($\sigma = 0.2$) or extremely large ($\sigma = 20$) as predicted. The EERs for $\sigma = 1$, $\sigma = 2$ and $\sigma = 10$ are similar to each other. Statistically, σ represents the standard deviation and controls the width of a Gaussian distribution. Three zero-mean Gaussian distribution with σ of 0.2, 0.5 and 20 are shown in Figure 4.4. As discussed before, the numerator of the exponent in (4.5.1) and (4.5.2) always ranges from 0 to 1. When $\sigma = 0.2$, a minor change in the input value results in a large difference in the output, making the kernel function too sensitive to the similarity score; on the other hand, when $\sigma = 20$, the output values corresponding to input value of 0 and 1 are almost identical, creating a constant Gram matrix regardless of the similarity score. Compared to the two cases above, $\sigma = 0.5$ has more adequate and healthier input (similarity score) to output (Gram matrix) ratio and thus, yielding better performance.

In the experiments, the exhaustive search approach is used to determine the number of retained principal components, N_p so that the recognition accuracy is optimized. From Figure 4.3, it is notable that the EER is at the lowest when N_p is between 100 to 150. For example when $\sigma = 0.5$, the lowest EERs for FVC2002 DB1 and FVC 2004 DB1 are 0.20% and 4.99% respectively at $N_p = 150$, while the lowest EERs for FVC2002 DB2 and FVC2004 DB2 are 1.33% and 7.38% respectively at $N_p = 100$. Since the principal components are input variables ordered in such a way that the first components account for as much variability in the data as possible, if the number of retained components is too small, there is insufficient relevant information taken into account. On the contrary, if too many components are kept, the analysis may include noisy and trivial components.

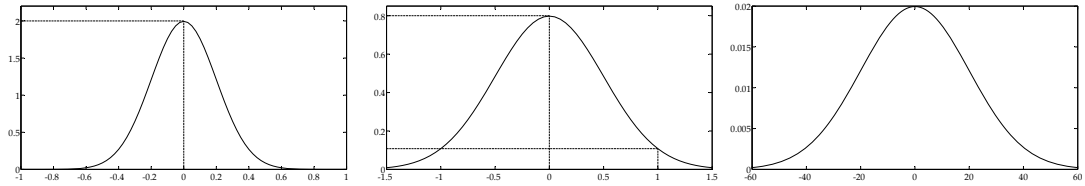
Besides the exhaustive search approach, the most common rule for choosing the number of principal components is the Kaiser-Guttman criterion [211–213]. It is a stopping rule for PCA which states that only components with eigenvalue of greater than one should be retained to best summarize the input variables. Following this criterion, the



(a) FVC2002 DB1. **(b)** FVC2002 DB2.

(c) FVC2004 DB1. **(d)** FVC2004 DB2.

Figure 4.3: EERs of the proposed fixed-length representation of MLC via KPCA-based S2V transformation while altering N_p and σ. The shaded areas represent the lowest EER regions.

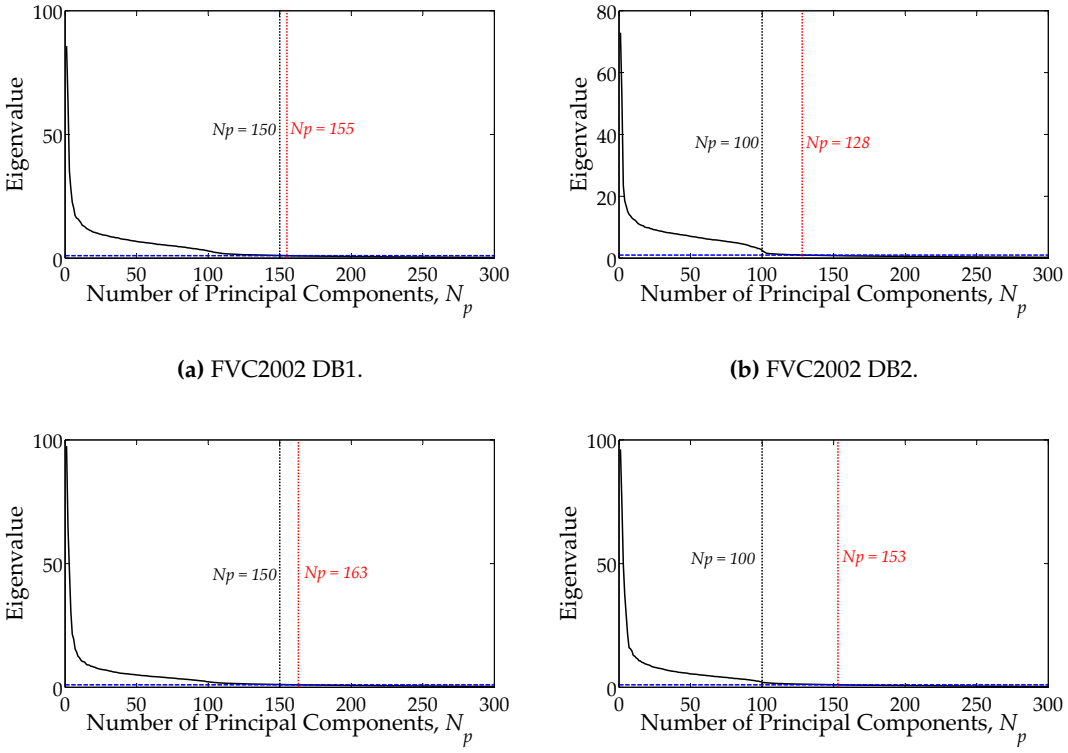


(a) $\sigma = 0.2$. **(b)** $\sigma = 0.5$. **(c)** $\sigma = 20$.

Figure 4.4: Gaussian distributions with zero mean and three different σ values.

numbers of principal components chosen for the four datasets are 155, 128, 163 and 153 respectively, as observed from the scree plots in Figure 4.5.

Furthermore, the scree plots help in visualizing the relationship between eigenvalues and the number of principal components. Another strategy is to examine the trend of the scree plot and find the point where it stops to descend precipitously and levels out eventually [214]. By this, the numbers of principal components taken are approximately 120, 104, 114 and 108 for the four datasets in order. This estimation agrees more to the actual results by exhaustive search compared to using the Kaiser-Guttman criterion.



(a) FVC2002 DB1.

(b) FVC2002 DB2.

(c) FVC2004 DB1.

(d) FVC2004 DB2.

Figure 4.5: Scree plots for the proposed KPCA method on different fingerprint datasets with $\sigma = 0.5$. The *blue horizontal dotted lines* in the plots indicate the Kaiser-Guttman criterion; the *black vertical dotted lines* and the *red vertical dotted lines* mark the number of principal components chosen according to exhaustive search and the Kaiser-Guttman criterion respectively.

4.6.3 Verification Rate of Cancellable Fixed-Length Representation

The results in this section correspond to the second (or final) outcome in Figure 4.2, where RP is applied for cancellable transformation. For the convenience of comparison, the experiments in this section use the same set of parameters for all four datasets, i.e. $\sigma = 0.5$ and $N_p = 125$ (an average between 100 and 150).

The manipulated parameter in this experiment is the reduced vector dimension after RP, D_r. From Figure 4.6, it is obvious that the EER increases as D_r decreases. This agrees with the observation in section 3.6.3, for cancellable MLC template. There is bound to have loss of information when the original data space is compressed into a much lower dimension. What is different from the results in section 3.6.3 is that the performance of the cancellable fixed-length representation is able to surpass that of before RP. The cause of the difference observed lies with the data distribution of the fingerprint template prior RP. The elements in the MLC template are sparsely distributed while the

KPCA-generated vector is normally distributed. RP has been proven to preserve information in normally distributed data [175], but does not work as well for unique distribution like MLC (section 3.6.3). Furthermore, unlike direct vectors matching, the matching of two unordered and variable-size templates (refer to section 3.5) is complex and might not be exactly adaptable to RP.

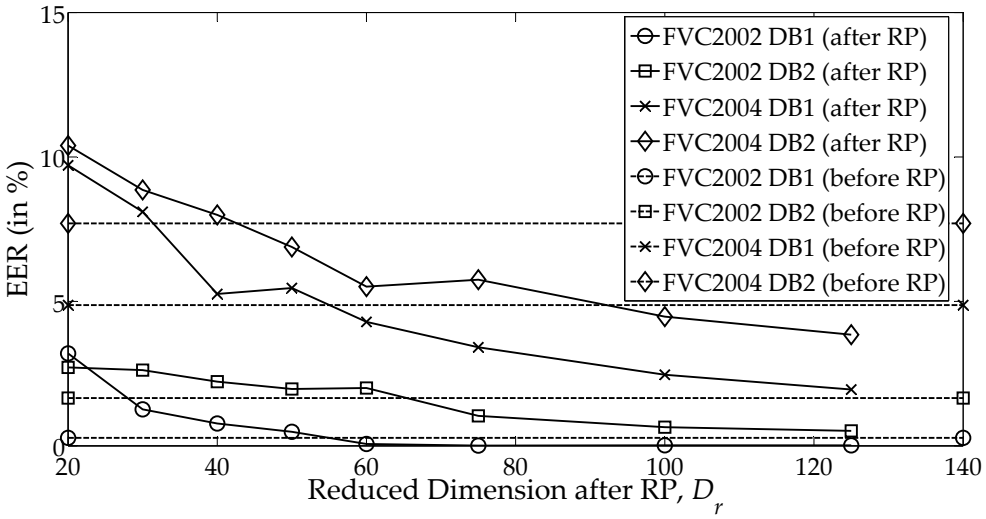


Figure 4.6: EERs of the proposed cancellable fixed-length fingerprint representation via KPCA-based S2V transformation and RP under stolen-key scenario.

In Figure 4.6, most plots have EER approximates to the original EER when $40 < D_r < 60$, except for FVC2002 DB1, which achieves equivalent EER when $60 < D_r < 80$. The EERs continue to drop tremendously afterwards, especially for the two FVC2004 datasets. Taking $D_r = 60$ as an example, the EERs for the four datasets are 0.51%, 2.00%, 4.29% and 5.51% respectively. These results correspond to the stolen-key scenario. The genuine-key scenario on the other hand, is able to achieve 0% EER for all datasets.

Table 4.2 shows the performance of the final outcome corresponding to $\sigma = 0.5$, $N_p = 125$ and $D_r = 60$. It is noteworthy that there is a great improvement in the EER compared to the original MLC. It proves the information preservation ability of the S2V transformation in the context of minutiae-based fingerprint template. Not only that, KPCA projects the templates onto a non-linear and orthogonal subspace so that the templates of different users are well-separated and thus, is able to enhance the verification accuracy of the final output. Figure 4.7 visualizes the improvement in performance by comparing the genuine-impostor distributions. The overlapping area between the distributions is greatly reduced, indicating a better genuine-impostor separation. Be-

sides, the proposed method outperforms all the other benchmarking cancellable fingerprint templates.

Table 4.2: Summary of the recognition accuracy (in terms of EER) of the proposed fixed-length cancellable fingerprint template compared to other existing methods.

Algorithm	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
Proposed method				
Original MLC (before S2V transformation, Chapter 3)	2.83	2.25	9.16	8.89
KPCA-based method (after S2V transformation)	**0.51**	2.00	**4.29**	**5.51**
Existing S2V transformation methods (both cancellable and non-cancellable)				
Nagar et al.[1] [108]	-	3.00	-	-
Bringer and Despiegel[2] [47]	-	1.70	-	-
Vij and Namboodiri[2] [197]	1-2	**1-2**	7-8	8-9
Nandakumar[2] [137]	0.80	0.70	-	-

[1]cancellable method.

[2]non-cancellable method.

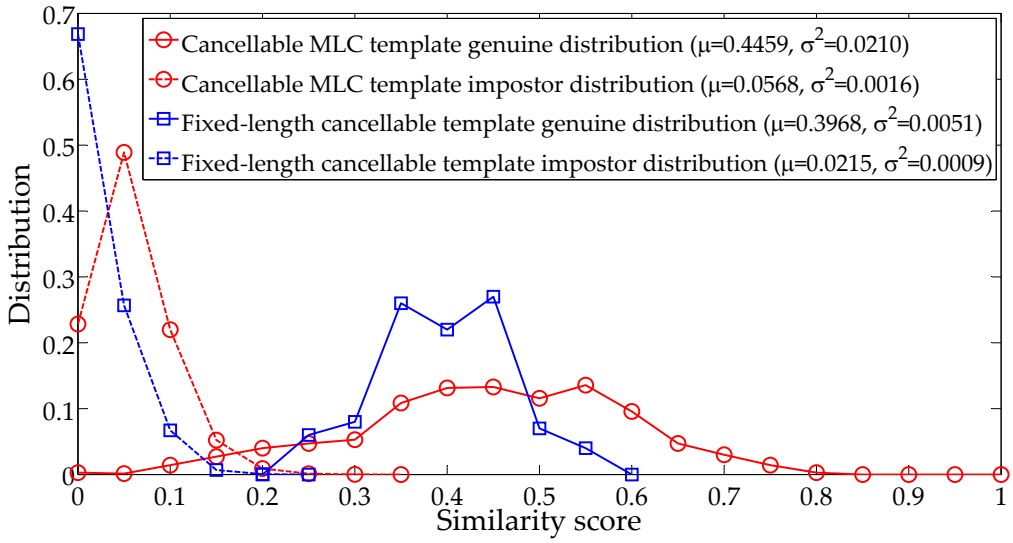


Figure 4.7: Comparison of genuine-impostor distribution between original MLC template and the proposed fixed-length representation for FVC2002 DB1.

4.6.4 Security and Privacy Analyses

The security and privacy of the proposed cancellable fixed-length fingerprint template is evaluated the same way as the MLC template in section 3.6.4, that is, from the aspect of non-invertibility, unlinkability and entropy of the final vector.

Non-invertibility: Resistance against Reverse Attack

Unlike the MLC template, the KPCA-based fixed-length representation is non-sparse. Without the sparseness constraint, the ℓ^1-minimization problem in (3.6.4) has to be replaced with ℓ^2-minimization. Given that the adversary has the knowledge of the random projection matrix ($\mathbf{R} \in \mathbb{R}^{N_p \times D_r}$), recovering the original vector ($\mathbf{V}_{\text{KPCA}} \in \mathbb{R}^{N_p}$) from the reduced vector ($\hat{\mathbf{V}}_{\text{KPCA}} \in \mathbb{R}^{D_r}$) is equivalent to solving the optimization problem:

$$\min \|\mathbf{V}_{\text{KPCA}}\|_2 \quad \text{subject to} \quad \mathbf{R}\mathbf{V}_{\text{KPCA}} = \hat{\mathbf{V}}_{\text{KPCA}}; \tag{4.6.2}$$

Since $D_r < N_p$, the equation above represents an underdetermined linear system and there are infinite solutions for $\mathbf{V}_{\text{KPCA}}$. Even if the training samples are revealed, the adversary would not know which templates belong to the user, so it does not provide additional condition for solving the equation.

For the S2V transformation phase, as only the first N_p principal components are chosen, the projection onto these principal components also results in dimensionality reduction. Therefore, it gives the adversary another optimization problem similar to (4.6.2) to solve. Provided that the Gram matrix can be exactly recovered (which requires immensely large number of trials through the two stages of optimization problems), the adversary can guess which training samples belong to the genuine user by observing the values of the Gram matrix. As the Gram matrix is essentially a matrix of similarities, the higher the value of an element in the matrix, the higher the possibility that the corresponding training sample belongs to the user.

Besides guessing the user's training samples from the Gram matrix, the adversary could also generate the cancellable templates corresponding to the stored training samples and find the one(s) most similar to the stolen template. Even so, the training samples used for KPCA are MLC templates, thus protected by the MLC algorithm. Refer to section 3.6.4 for the non-invertibility analysis of the MLC algorithm. With the multi-

layer protection, including the MLC algorithm, S2V transformation (KPCA) and RP, the raw minutiae information is kept secret from the adversary.

Unlinkability: Resistance against Linkage Attack

The unlinkability test is performed by computing the separability between templates of the same fingerprint with multiple revocation attempts as explained in section 3.6.4. However, since the fingerprint template is now ordered and fixed-length, the number of matchable minutiae pairs is replaced with the similarity score between two vectors. Also, training is required for the proposed KPCA-based S2V transformation, so the unlinkability test follows the procedure below:

1. Randomly choose six out of eight samples per fingerprint for KPCA training. The remaining two are used for testing.
2. Generate five versions of fixed-length cancellable templates as the enrolled templates from the first testing sample using five distinct revocation key.
3. The second testing sample is assigned another ten keys to produce ten unique query templates.
4. Match the query templates against the enrolled templates of the same fingerprint and generate the distribution of similarity scores between the templates. The separability is calculated based on (3.6.5).
5. Repeat the steps above fivefold with different random samples to obtain an average separability measure.

In this experiment, $D_r = 60$ is used. Table 4.3 shows the separability between the same-key distribution and the different-key distribution for the KPCA-based cancellable template. Although the different-key genuine matching does not produce all-zero scores like that of the variable-size MLC template (refer to Table 3.10), the separability values are slightly higher. This is because the same-key genuine distribution has very small variance while the means of the two distributions remain well-separated. Figure 4.8 compares the distributions of the dataset with the best and the worst separability. It is easily observed that the higher the separability is, the less the intersected area between the two distributions.

Table 4.3: Separability of the proposed cancellable fixed-length representation expressed in the form of "separability$(\mu_{SKG}, \sigma^2_{SKG})[\mu_{DKG}, \sigma^2_{DKG}]$". μ_{SKG} and σ^2_{SKG} represent the mean and variance of the same-key genuine matching distribution, while μ_{DKG} and σ^2_{DKG} are the equivalent parameters of the different-key genuine matching distribution. Since the decimal values shown are rounded to the nearest 0.01, any value that is less than 0.005 are written as <0.005.

FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
4.59 (0.66,0.01) [0.29,<0.005]	3.53 (0.68,0.02) [0.29,<0.005]	3.09 (0.57,0.01) [0.29,<0.005]	3.21 (0.59,0.02) [0.29,<0.005]

Table 4.4: Entropy (in bits) of the proposed cancellable fixed-length representation of fingerprint. The first number represents the average discrete entropy per vector component and the second number represents the total discrete entropy of the vector. The number in parenthesis is the average differential entropy per component.

FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
5.14, 308.13 (-1.44)	5.14, 308.43 (-1.44)	5.14, 308.39 (-1.44)	5.14, 308.46 (-1.44)

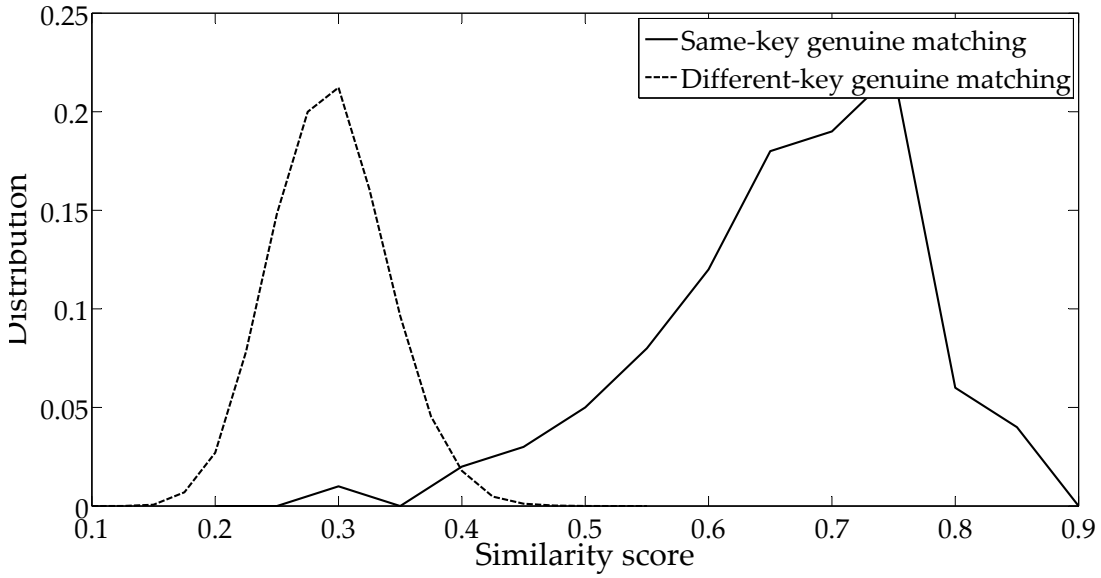
Entropy: Resistance against Brute Force Attack

Again, the procedure of calculating the entropy follows the formulae in (3.6.6) and (3.6.8) from section 3.6.4. What is different is that the output is now fixed-length and ordered, so the entropy of each feature component can be calculated separately and the total entropy of the entire template is the sum of the entropies of all the components, that is,

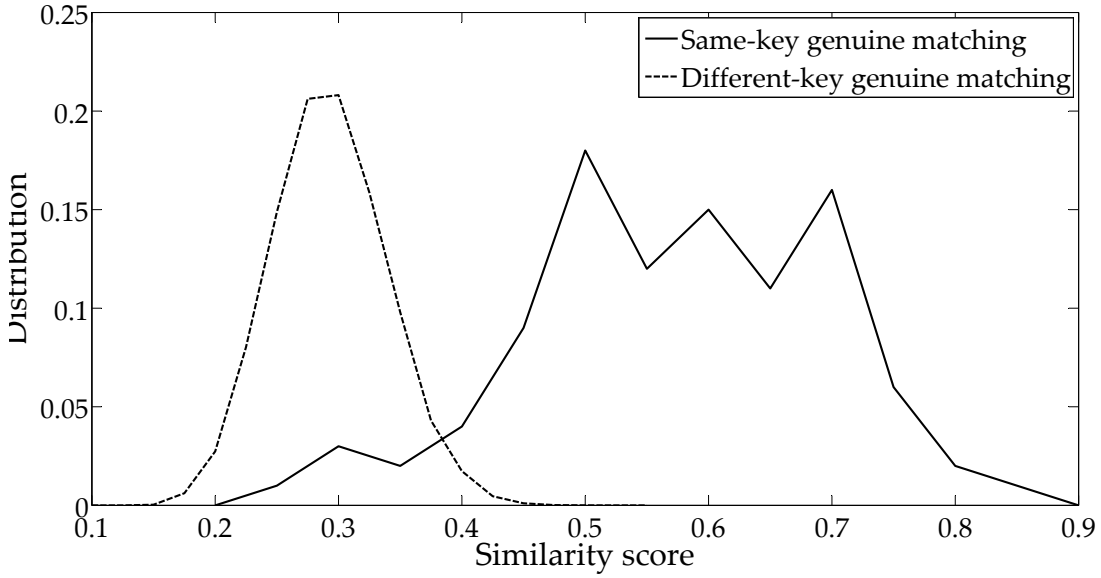
$$H(\hat{\mathbf{V}}_{\text{KPCA}}) = \sum_{i=1}^{D_r} H(\hat{V}_{\text{KPCA}(i)}), \tag{4.6.3}$$

where $H(\hat{V}_{\text{KPCA}(i)})$ is the entropy of the ith component of the fingerprint template as defined in (3.6.6) and (3.6.8).

From Table 4.4, the entropies of all datasets are similar to each other, with 308 bits of total discrete entropy and 5.14 bits of average entropy per vector component. Although the total entropies approximate to the ones obtained for cancellable MLC template in section 3.6.4, the average entropies per component are much higher. The sparseness of MLC affects the values in the RP-ed template and thus, the entropy is low even after RP. On the other hand, the fixed-length cancellable template originated from a normally distributed KPCA-based vector, hence yield higher entropy per component. The differential entropies are negative due to small variance of the distribution of the feature components.



(a) FVC2002 DB1 (highest separability).



(b) FVC2004 DB1 (lowest separability).

Figure 4.8: Examples of same-key and different-key distributions.

4.6.5 Computational Complexity

The proposed S2V transformation via KPCA consists of a training stage and a transformation stage. The computational complexity of each of the stages are evaluated individually.

Preliminarily, the matching score calculation between two MLC templates yields time complexity of $\mathcal{O}(N_m^2)$. Although two fingerprints may have different number of minutiae, the general term N_m is used to represent the average number of minutiae in any

fingerprint. In the KPCA training stage (Algorithm 4.1), the matching algorithm is executed $\frac{N_t}{2}[2(1) + (N_t - 1)(1)] - N_t = \frac{1}{2}(N_t^2 - N_t)$ times, so the time complexity of the kernel construction process is $\mathcal{O}(\frac{1}{2}(N_t^2 - N_t)N_m^2)$, where N_t is the total number of training samples. If $N_t \gg 1$, the expression can be simplified to $\mathcal{O}(N_t^2 N_m^2)$. Other than kernel calculation, the training process also includes eigenvalue decomposition and the complexity is $\mathcal{O}(N_t^3)$. Besides the main algorithm of the S2V transformation, a complete training process also includes the MLC generation of the N_t training samples. Therefore, the time complexity of the training stage is $\mathcal{O}(\max(N_t^2 N_m^2, N_t^3, N_t N_m D_m))$, where $\mathcal{O}(N_m D_m)$ is the complexity of the MLC generation step (refer to section 3.6.5).

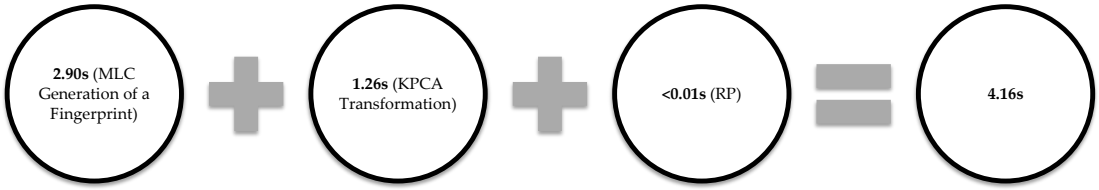
As for the transformation stage, the matching is executed N_t times and the complexity of kernel construction is $\mathcal{O}(N_t N_m^2)$. Additionally, the principal components projection step has complexity of $\mathcal{O}(N_t N_p)$. It is known from section 3.6.5 that the complexity of RP for the vectorized template is $\mathcal{O}(N_p D_r)$. To conclude all, the time complexity of the entire fixed-length cancellable template generation process is $\mathcal{O}(\max(N_t N_m^2, N_t N_p))$. The term $\mathcal{O}(N_p D_r)$ is negligible because $N_t > N_p > D_r$.

From the aspect of CPU runtime, the average training time over four datasets is 3139.78s and the average time needed for a single template generation is 5.55s. While the number of training samples used is the same for all datasets ($N_t = 600$), the average number of minutiae (N_m) and the MLC dimension (D_m) differ among the datasets. For instance, since FVC2004 DB2 has the least average number of minutiae (Table 3.2) and the lowest MLC dimension (Table 3.3 and 3.7), it has the shortest runtime for both stages. Figure 4.9 depicts the breakdown chart of CPU runtime for FVC2004 DB2. In the training stage, the MLC generation phase takes more than double the time of KCPA training. Similarly in the template generation stage, MLC generation for a single fingerprint takes longer than transforming it into a fixed-length vector. Plus, the time taken for RP is trivial. Therefore, the observation shows that the MLC generation phase is the bottleneck of the whole process.

Table 4.5: CPU runtime of the proposed cancellable fingerprint template generation scheme, running on MATLAB environment (Windows 7) with an Intel® Core™ i5-2430M 2.40GHz processor.

Stage	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
Training	3083.73s	3649.99s	3325.85s	2499.56s
Template Generation	5.88s	6.60s	5.54s	4.16s

(a) Training stage.



(b) Template generation stage.

Figure 4.9: CPU runtime breakdown chart of training stage and template generation stage for FVC2004 DB2.

The training algorithm may seem complex and time-consuming at first glance, but it does not affect the real-life applicability of the proposed scheme as it is processed off-line. With high-speed computers available nowadays and with more efficient computing platform, the on-line template generation process can definitely be much shorter than what is shown in Table 4.5.

4.7 Summary

A S2V transformation technique through kernel subspace analysis has been proposed in this chapter. Among the subspace analysis models, PCA was chosen to be incorporated with kernel method to realize the task due to its adaptability to the minutiae data. For this purpose, a unique kernel function has been introduced and verified. The KPCA-based method can be separated into the training stage and the transformation stage. The former performs analysis on the training samples to obtain the principal components while the latter transforms the variable-size and unordered input data into a fixed-length and ordered vector.

In the experiments, the proposed method was implemented in a complete fingerprint template generation scheme, consisting of the MLC generation phase, the KPCA-based S2V transformation phase and the cancellable transformation phase via RP. A great improvement in the verification accuracy was observed after applying the proposed S2V transformation due to the good data separation introduced by KPCA.

Furthermore, the fixed-length cancellable template is well-protected by the non-sparse RP step. Although the training samples stored for KPCA reveals the original variable-size template (MLC), the samples are not labelled. The adversary is required to attack the Gram matrix first before knowing which training samples belong to the genuine user. In addition, the cracking the MLC template is another challenge for the adversary. Such multi-layer protection provides high privacy to the users' fingerprint data. Other aspects of analysis such as unlinkability and entropy also show better security compared to the original MLC template.

Even though the S2V transformation is an additional phase to the previously proposed cancellable template generation scheme (Chapter 3) and results show that requires additional CPU runtime, it poses no complication to the applicability of the proposed scheme in real-life applications. This is because, the most time-consuming steps (training) are processed off-line and the on-line operations can be done effectively with current computer technology.

Overall, the proposed S2V transformation via KPCA can be considered as an enhancement to the existing scheme from the aspect of performance, security and privacy, and most importantly, the applicability of the final outcome in various biometrics or pattern classification systems (note that the motivation of transforming unordered set to ordered vector is to enable the use of minutiae-based fingerprint templates in bio-cryptosystems, continuous classification, and sophisticated classifiers and template binarization techniques as discussed in section 4.1).

CHAPTER 5

S2V Transformation via Bag-of-Minutiae Modelling

5.1 Background

In addition to the S2V transformation technique presented in the previous chapter, another alternative to transform unordered set to ordered vector is proposed in this chapter, namely bag-of-minutiae (BoM) modelling. The BoM model is derived from the bag-of-words (BoW) model. It represents a minutiae set by the occurrence of certain minutia prototypes in the original set. Since two analogous minutiae might not have the exact same value, the prototypes serve as a guideline of partitioning the minutiae space. Besides text classification, the BoW model has also been implemented in many image classification problems [215, 216], which is also known as the bag-of-visual-words model. The keypoint detectors commonly used in bag-of-visual-words modelling for feature extraction include Laplacian of Gaussian (LoG), scale-invariant feature transform (SIFT), Hessian Laplace and Hessian Affine. However, these features do not produce comparable performance as minutiae-based methods do in fingerprint recognition [85].

The related work for S2V transformation or fixed-length representation of fingerprints have been presented in section 4.2, so it is omitted in this chapter. The proposed BoM modelling essentially belongs to the histogram-based approach in the literature categorization. In this chapter, BoM modelling is demonstrated through both K-means clustering (hard quantization) and dictionary learning (soft quantization). Some parts of the results obtained in this chapter has been published in [217].

5.2 Preliminaries: Dictionary Learning for Soft Quantization

Given a training set $\mathbf{\Omega}_{\text{train}} \in \mathbb{R}^{D_m \times N_t}$ where D_m is the dimension per minutia and N_t is the number of minutiae in the training set, the aim of dictionary learning is to partition the feature space so that the following general objective function is fulfilled:

$$\min_{\mathbf{C},\mathbf{B}} \|\mathbf{\Omega}_{\text{train}} - \mathbf{CB}\|_2^2 \text{ s.t. } \forall i, \|\mathbf{B}_{(i)}\|_0 \leq \lambda, \tag{5.2.1}$$

where $\|\cdot\|_0$ and $\|\cdot\|_2$ indicates the ℓ^0 norm and the ℓ^2 norm of vectors respectively, $\mathbf{C} \in \mathbb{R}^{D_m \times K}$ is the dictionary containing K atoms with each corresponding to one partition, $\mathbf{B} \in \mathbb{R}^{K \times N_t}$ is the resulting sparse representation, $\mathbf{B}_{(i)}$ is the ith column of $\mathbf{B}$ and λ, also known as the target sparsity, indicates the maximum number of atoms a minutia may be assigned to. The dictionary learnt is usually used for sparse approximation. In general, dictionary learning is an iterative process with each iteration composes of a sparse approximation step and a dictionary update step, which are equivalent to minutiae labelling and space partitioning in section 5.4. The sparse approximation step deduces the sparse representation of $\mathbf{\Omega}_{\text{train}}$ corresponding to the dictionary obtained from the previous iteration, while the dictionary update step re-evaluates the dictionary based on the sparse representation. In this section, we discuss the orthogonal matching pursuit (OMP) for sparse approximation and K-singular value decomposition (K-SVD) as the dictionary update algorithm.

5.2.1 Sparse Approximation: OMP

Matching pursuit is a greedy approach towards finding the sub-optimal solution to the problem described in (5.2.1) given the dictionary, $\mathbf{C}$. It is an iterative process which removes the selected atoms from the residual at each iteration until the target sparsity is reached and thus, increases the computational efficiency. The residual is used to compute the new coefficients by projecting it onto the atoms in the dictionary. OMP [179] is a popular variant of matching pursuit, in which one atom can only be selected once so that the projection is orthogonal. The pseudo-code of the OMP algorithm is presented in Algorithm 5.1.

Algorithm 5.1: OMP

Data: $\mathbf{\Omega}, \mathbf{C}, \lambda$
Result: $\mathbf{C}$
begin
  $N_m \longleftarrow$ size($\mathbf{\Omega}$,1)
  $K \longleftarrow$ size($\mathbf{C}$,2)
  for $i \leftarrow 1$ *to* N_m **do**
    $\mathfrak{R}_{(0)} \longleftarrow \boldsymbol{\omega}_{(i)}$ `// initialize residual`
    $\mathbf{\Phi} \longleftarrow \varnothing$ `// selected atoms`
    for $j \leftarrow 1$ *to* λ **do**
      $j \longleftarrow \arg\max_{k \in [1,K]}(|\langle \mathfrak{R}_{(j)}, \mathbf{C}_{(k)} \rangle|)$
      $\mathbf{\Phi}_{(j)} \longleftarrow \mathbf{\Phi}_{(j-1)} \cup \mathbf{C}_{(k)}$
      $\mathbf{B}_{(i)} \longleftarrow \mathbf{\Phi}_{(j)}(\mathbf{\Phi}_{(j)}^{\mathrm{T}}\mathbf{\Phi}_{(j)})^{-1}\mathbf{\Phi}_{(j)}^{\mathrm{T}}\mathfrak{R}_{(j-1)}$
      $\mathfrak{R}_{(j)} \longleftarrow \mathfrak{R}_{(j-1)} - \mathbf{B}_{(i)}$
    end
  end
end

5.2.2 Dictionary Update: *K*-SVD

K-SVD can be seen as a generalized framework of *K*-means algorithm [218]. Within one dictionary learning iteration, the *K*-SVD algorithm consists of K sub-iterations with each corresponding to one atom in the dictionary. At the ith sub-iteration ($i \in [1, K]$), the ith column of the dictionary, $\mathbf{C}_{(i)}$ and the ith row of the sparse representation, $\mathbf{B}_{(i\sim)}$ are removed from the matrices, and the residual matrix is computed as $\boldsymbol{\varepsilon} = \mathbf{\Omega}_{\text{train}} - \sum_{j=1}^{K\setminus\{i\}} \mathbf{C}_{(j)}\mathbf{B}_{(j\sim)}$. Let $\boldsymbol{\varepsilon}_{(i)}$ be the i column of $\boldsymbol{\varepsilon}$, SVD is then computed based on $\boldsymbol{\varepsilon}_{(i)}\mathbf{B}_{(i\sim)} = \mathbf{E}\Delta\mathbf{F}^{\mathrm{T}}$. The updated ith atom is calculated as $\mathbf{C}_{(i)} = \mathbf{E}\mathbf{F}^{\mathrm{T}}$. In a nutshell, the *K*-SVD algorithm performs SVD K times in each iteration to update the entire dictionary. Therefore, the computational power required is high. Besides, the approximation accuracy is unsatisfactory as global optimum is not guaranteed [219]. Several variants of *K*-SVD were proposed to improve the performance of the dictionary learnt such as: i) enhanced *K*-SVD (E*K*-SVD) [220] which finds the optimal dictionary size given a dataset; ii) label-consistent *K*-SVD (LC-*K*SVD) [221] which incorporates the concept of supervised learning; and iii) immune *K*-SVD (I*K*-SVD) [222] which introduces the immune mechanism to improve the characteristics of global minimum.

5.3 Nomenclature

Symbol	Description
$\mathbf{\Omega}$	MLC template of a fingerprint
$\mathbf{\Omega}_{\text{train}}$	training minutia vectors (MLCs) for BoM modelling
D_m	dimension per minutia vector
N_t	total number of training samples
K	dictionary size for *a posteriori* BoM modelling, also the dimension of BoM-generated fingerprint vector
$\mathbf{C}$	dictionary trained for *a posteriori* BoM modelling
$\mathbf{B}$	sparse representation of the input minutia vectors (MLCs)
I	number of iterations for the assignment/update process
λ	sparsity target for dictionary learning
$\mathbf{V}_{\text{BoM}} \in \mathbb{R}^K$	fixed-length fingerprint template generated through BoM modelling
$\hat{\mathbf{V}}_{\text{BoM}} \in \mathbb{R}^{D_r}$	cancellable fixed-length fingerprint template generated through BoM modelling
D_r	dimension of the cancellable fixed-length template
N_φ	number of orientation levels in MLC construction
l	length of lines for MLC construction
N_l	number of lines for MLC construction
d	distance between two sample points
r	radius of circles centring at the sample points

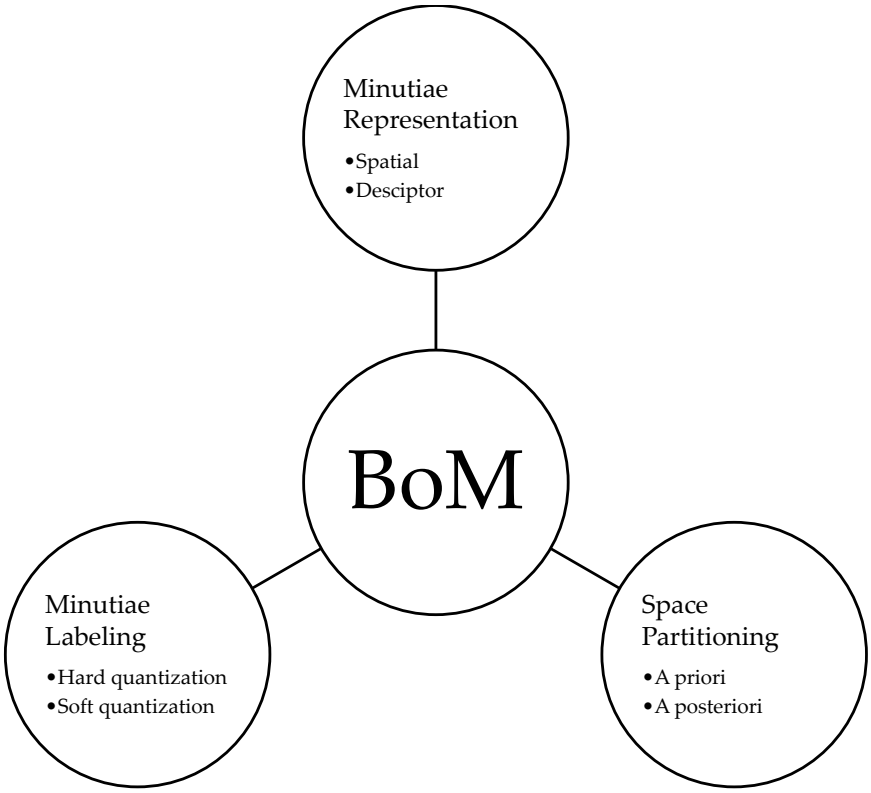


Figure 5.1: Three design choices of the BoM model.

5.4 Proposed BoM Modelling

5.4.1 The BoM Model

Bianconi and Fernández [223] discussed the five dichotomous design choices of BoW modelling for image classification but not all are applicable in the context of minutiae-based fingerprint recognition. In this section, we explain the three design approaches of the proposed BoM model, each with two alternative operational modes as illustrated in Fig. 5.1.

Minutiae Representation

The choice of minutiae representation refers to the feature space which BoM operates on. It can be either the three-dimensional spatial representation or the descriptor-based representation. The former directly utilizes the ISO template of minutiae (except for the minutia type) in space partitioning; whereas the latter transforms the minutiae into the descriptor space prior partitioning.

The spatial representation of a minutia indicates the location of the minutia in the fingerprint image and its ridge orientation. The minutiae aggregations approach [107,108] is one example of creating clusters based on this representation. Since translation and rotation often occurs in fingerprint images and are unpredictable, the drawback of using such representation is that the partitions should move accordingly, and thus are impossible to define. One solution to this issue is to perform fingerprint pre-alignment so that the fingerprints are properly aligned with the space where the partitions are constructed on. Although the vast advancement in the current computer technology has set aside the concern over high computational time of fingerprint pre-alignment, the errors in fingerprint pre-alignment can affect the accuracy of the transformed fingerprint template.

On the other hand, the descriptor-based representation transforms the minutiae into a translation- and rotation-invariant local space and consequently eliminates the need for fingerprint pre-alignment. Descriptors employed for set-to-vector transformation in the literature are the fixed-radius vicinity [47] and minutiae triplets [100,197], among which the fixed-radius descriptor is more robust against local non-linear distortions. However, descriptors of higher complexity such as minutia cylinder-code (MCC) [38] and MLC (Chapter 3) are more suitable in BoM as the dimension of the minutia vectors generated is adjustable.

Space Partitioning

Two alternative options of partitioning the feature space include *a priori* and *a posteriori*. *A priori* partitioning is independent of data and the resulting partitions are usually user-defined. For example, the random cuboid clustering technique [107, 108] creates randomly chosen partitions of different sizes on the feature space regardless of the minutiae distribution. As the minutiae are not uniformly distributed, redundant and overpopulated partitions are unavoidable. On the contrary, *a posteriori* partitioning requires the knowledge of data distribution and the partitions are learnt from the preliminary data. The clusters selection technique used by Bringer and Despiegel [47] and Farooq et al. [100] demonstrate the *a posteriori* partitioning, in which cluster centroids are selected from the training data through certain criteria. A more advanced technique was used by Vij and Namboodiri [197] where the partitions are learnt through K-means algorithm. The proposed BoM model is in fact a generalized set-to-vector transforma-

tion framework of that presented in [197]. Other algorithms such as various supervised and unsupervised learning algorithms are also possible approaches towards *a posteriori* partitioning.

Minutiae Labelling

Minutiae labelling refers to the process of assigning minutiae to the most appropriate partition(s). The options are hard labelling (or hard quantization) and soft labelling (or soft quantization). The former assigns each minutia to exactly one partition only; while the latter allows a minutia to be associated with multiple partitions. All existing methods [47, 100, 107, 108, 197] adopts the concept of hard labelling.

Besides *K*-means clustering, other possible hard labelling techniques include *K*-means++, *K*-medians clustering and self-organizing map. The output of these labelling techniques is binary — a minutia either belongs a partition or not. Albeit computationally simple, hard labelling is sensitive to noisy data and outliers. Besides, it is not able to handle two or more highly overlapping classes. As opposed to that, soft labelling introduces fuzziness to the partition borders so that close-to-border minutiae may be assigned to more than one partition. By introducing tolerance in minutiae assignment, the quantization error may be reduced. Soft labelling produces a real-numbered weight indicating the strength of association between a minutia and a partition as the output. As the minutiae obtained for the same fingerprint are usually different (either translated, rotated, or missing), such weight value helps to reduce the error caused by mislabelling a minutia when distortions occur. A few instances of soft labelling include fuzzy *C*-means clustering, sparse approximation and Gaussian mixture modelling.

5.4.2 S2V Transformation using the BoM Model

In this thesis, we demonstrate two realizations of the BoM model, each from one design option of minutiae labelling, i.e. hard quantization and soft quantization. Hard quantization is implemented with *K*-means clustering while it's contra is realized by using dictionary learning. Both of them belongs to *a posteriori* partitioning which requires training for dictionary prior the transformation. The training stage and the transformation stage are detailed below:

Algorithm 5.2: Hard quantization training

Data: $\mathbf{\Omega}_{\text{train}}, K, I$ `// I is the number of iterations`
Result: $\mathbf{C}$

begin
  $\mathbf{C} \longleftarrow K$ minutia vectors randomly chosen from $\mathbf{\Omega}_{\text{train}}$
  for $i \leftarrow 1$ *to* I **do**
    $\mathbf{B} \longleftarrow$ knnsearch$(\mathbf{C}, \mathbf{\Omega}_{\text{train}})$ `// assignment`
    for $j \leftarrow 1$ *to* K **do**
      $\mathbf{C}_{(j)} \longleftarrow$ centroid$(\mathbf{\Omega}_{\text{train}(\mathbf{B}==j)})$ `// dictionary update`
    end
  end
end

Training Stage

Regardless of the minutia labelling technique used, the training stage involves an iterative process of minutiae labelling and space partitioning. Given the training set $\mathbf{\Omega}_{\text{train}} \in \mathbb{R}_{\geq 0}^{D_m \times N_t}$ containing N_t minutiae vectors (MLC), the objective function in (5.2.1) can be rewritten as:

$$\min_{\mathbf{C},\mathbf{B}} \|\mathbf{\Omega}_{\text{train}} - \mathbf{CB}\|_2^2 \text{ s.t. } \forall i, \|\mathbf{B}_{(i)}\|_0 \leq \lambda, \tag{5.4.1}$$

where $\mathbf{C} \in \mathbb{R}^{D_m \times K}$ is the dictionary obtained and $\mathbf{B} \in \mathbb{R}^{K \times N_t}$ is the sparse representation of the training set.

For hard quantization, $\lambda = 1$ indicates that one minutia vector is assigned to only one partition. Algorithm 5.2 shows the pseudo-code of obtaining the dictionary via K-means clustering, or also known as Lloyd's algorithm [224]. The initial dictionary consists of minutia vectors randomly selected from the training set. Similar to the dictionary learning process of soft quantization discussed in section 5.2, an iteration consists of an assignment (or minutiae labelling) step and a dictionary update (or space partitioning) step. In the assignment step, each minutia vector in the training set, $\boldsymbol{\omega}_{\text{train}(i)}$ is assigned to a cluster in the updated dictionary based on the nearest neighbour algorithm:

$$\arg\min_{j \in [1,K]} \|\boldsymbol{\omega}_{\text{train}(i)} - \mathbf{C}_{(j)}\|_2, \tag{5.4.2}$$

where $\mathbf{C}_{(j)}$ denotes the jth cluster in the dictionary. Subsequently, the dictionary is updated by moving the cluster centroids to the centroids of the corresponding minutia vectors.

Algorithm 5.3: Soft quantization training

Data: $\mathbf{\Omega}_{\text{train}}, K, \lambda, I$
Result: $\mathbf{C}$
begin
  $\mathbf{C} \longleftarrow K$ minutia vectors randomly chosen from $\mathbf{\Omega}_{\text{train}}$
  for $i \leftarrow 1$ *to* I **do**
    $\mathbf{B} \longleftarrow \text{omp}(\mathbf{\Omega}_{\text{train}}, \mathbf{C}, \lambda)$ `// sparse approximation via OMP`
    for $j \leftarrow 1$ *to* K **do**
      $\Lambda \longleftarrow \{k | 1 \leq k \leq K, \mathbf{B}_{(k)} \neq 0\}$ `// active set`
      $\boldsymbol{\varepsilon}_{(j)} \longleftarrow \mathbf{\Omega}_{\text{train}} - (\mathbf{CB} - \mathbf{C}_{(j)}\mathbf{B}_{(\Lambda)})$
      $\boldsymbol{\varepsilon}_{(j)} \longleftarrow \mathbf{E}\Delta\mathbf{F}^{\text{T}}$ `// perform SVD`
      $\mathbf{C}_{(j)} \longleftarrow \mathbf{E}_{(1)}$ `// dictionary update`
    end
  end
end

On the other hand, soft quantization allows $\lambda > 1$. Consider the linear system

$$\mathbf{\Omega} = \mathbf{CB}, \tag{5.4.3}$$

dictionary learning for soft quantization is used to solve an underdetermined system in practice, where $D_m << K$. The dictionary learning process follows the algorithm described in Algorithm 5.3 and section 5.2.2.

Transformation Stage

Fig. 5.4 shows the procedure of the set-to-vector transformation. Each time a fingerprint is presented to the system, the MLC-based template $\mathbf{\Omega}$, together with the dictionary learned are used to compute for the approximated matrix $\mathbf{V} \in \mathbb{R}^{K \times N_m}$ according to nearest neighbour algorithm (for hard quantization) or OMP (for soft quantization). The final fixed-length representation of fingerprint is obtained by performing one of the following pooling functions:

i) sum-pooling:

$$\mathbf{V}_{\text{BoM}} = \left[\sum_{i=1}^{N_m} B_{(1i)}, \sum_{i=1}^{N_m} B_{(2i)}, ..., \sum_{i=1}^{N_m} B_{(Ki)} \right]; \tag{5.4.4}$$

ii) mean-pooling:

$$\mathbf{V}_{\text{BoM}} = \left[\frac{1}{N_m} \sum_{i=1}^{N_m} B_{(1i)}, \frac{1}{N_m} \sum_{i=1}^{N_m} B_{(2i)}, ..., \frac{1}{N_m} \sum_{j=1}^{N_m} B_{(Ki)} \right]; \tag{5.4.5}$$

Algorithm 5.4: BoM transformation

Data: $\mathbf{\Omega}$,method,pooling
Result: $\mathbf{V}_{\text{BoM}}$

```
begin
    if method == 0 then                          // perform hard quantization
        load C
        V ⟵ knnsearch(C, Ω)
    end
    else if method == 1 then                     // perform soft quantization
        load C, λ
        B ⟵ OMP(Ω, C, λ)                          // refer to Algorithm 5.1
    end
    if pooling == 0 then
        V_BoM ⟵ sumpool(B)                              // refer to (5.4.4)
    end
    else if pooling == 1 then
        V_BoM ⟵ meanpool(B)                             // refer to (5.4.5)
    end
    else if pooling == 2 then
        V_BoM ⟵ maxpool(B)                              // refer to (5.4.6)
    end
end
```

iii) max-pooling:

$$\mathbf{V}_{\text{BoM}} = \left[\max_{i \in [1,N_m]} B_{(1i)}, \max_{i \in [1,N_m]} B_{(2i)}, ..., \max_{i \in [1,N_m]} B_{(Ki)} \right]. \tag{5.4.6}$$

Max-pooling does not apply to hard quantization-generated matrix as it is a binary matrix (i.e. $\mathbf{B} \in \{0,1\}^{K \times N_m}$) and does not reflect the strength of belonging between the minutia vectors and the clusters. This makes up the five combinatorial BoM models that will be examined in the experiments, including *HQ+SUMPOOL*, *HQ+MEANPOOL*, *SQ+SUMPOOL*, *SQ+MEANPOOL* and *SQ+MAXPOOL*, in which *HQ* and *SQ* represent hard quantization and soft quantization respectively.

5.5 Experiments and Analyses

5.5.1 Testing Protocol

Similar to the KPCA-based S2V transformation technique, BoM modelling requires training data prior template generation. The cancellable template generation procedure depicted in Figure 4.2 is adopted, except that the S2V transformation technique

used is now BoM modelling. Since vectors generated by hard quantization are essentially histograms, histogram intersection is used as the similarity measure between two vectors:

$$S = \frac{\sum_{i=1}^{K} \min(V'_{\mathrm{BoM}(i)}, V''_{\mathrm{BoM}(i)})}{K}; \tag{5.5.1}$$

for soft quantization-generated vectors, the one minus normalized Euclidean distance as described in (4.6.1) is used for matching. $\mathbf{V}'_{\mathrm{BoM}}$ and $\mathbf{V}''_{BoM}$ are two instances of the proposed fixed-length fingerprint representation. Moreover, the cancellable templates after RP, $\hat{\mathbf{V}}_{\mathrm{BoM}}$ are matched with (4.6.1) regardless of the minutiae labelling option.

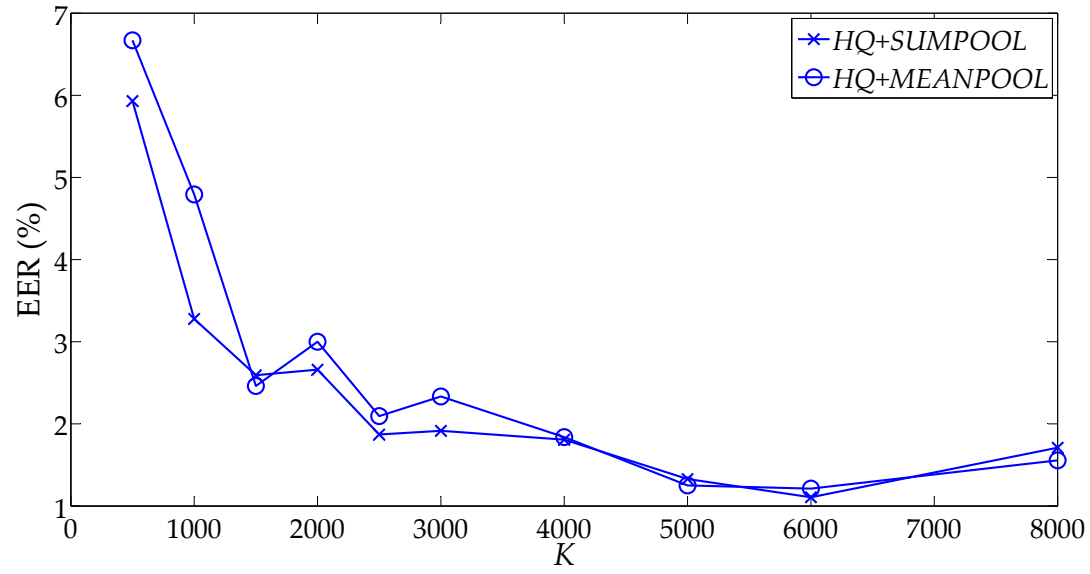
Also, a different set of parameters were used for the experiments in this chapter. The dimension for each minutia vector (D_m) generated using the best-performing MLC parameters suggested in the Chapter 3 ranges from 900 to 1476, depending on the dataset. Due to this, the dictionary size, K would have to be much larger ($D_m << K$), causing unduly fine quantization on the minutia vectors. Hence, the MLC parameters used in this chapter are: $N_\varphi = 6$, $N_l = 3$, $l = 240$, $d = 16$ and $r = 30$, resulting in $D_m = 576$, for all datasets. Now that the MLC parameters are the same for all datasets, the experiments in section 5.5.2 are performed on FVC2002 DB1 only to determine the optimal parameters of the S2V transformation.

5.5.2 Effect of Dictionary Size, Target Sparsity and Pooling Function

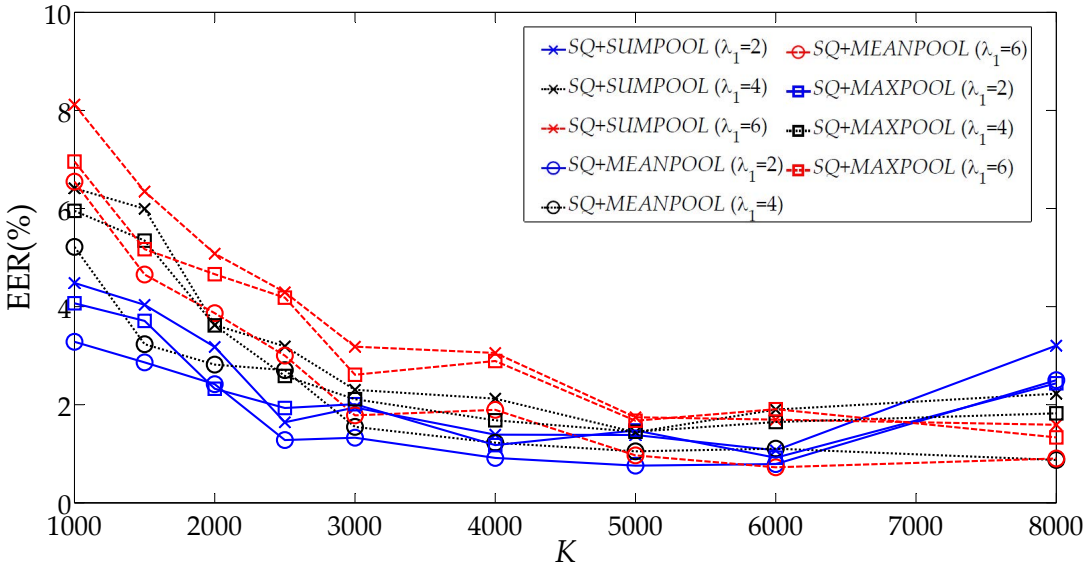
In this section, the effect of varying the parameters of the proposed S2V transformation on the system performance is studied. These parameters include the dictionary size, K and sparsity, λ (soft quantization only). The performances of different pooling functions are also investigated.

Figure 5.2 depicts the averaged EER over five trials of different methods and parameters. For hard quantization, the performance of sum-pooling and mean-pooling are similar to each other. The vectors generated by *HQ+MEANPOOL* are the normalized version of *HQ+SUMPOOL*-generated vectors. As histogram normalization does not change the shape of the histogram, the performance does not differ much.

As for soft quantization approach, although max-pooling has been shown to outperform other pooling functions in image classification [221, 225], our results show otherwise. In the plot, it is obvious that mean-pooling performs better than other pooling functions. In practice, max-pooling creates a better separation between foreground



(a) Hard quantization (*HQ*).



(b) Soft quantization (*SQ*).

Figure 5.2: Performance of the proposed minutiae-based fingerprint template after S2V transformation via BoM with different parameter values. The results correspond to FVC2002 DB1 dataset.

(genuine) distribution and background (impostor) distribution. Unlike mean-pooling, max-pooling only captures the maximum over the pool and thus, does not result in smoothing effect that causes inter-class variability to decrease. Besides, it is more robust against local spatial variations compared to mean-pooling. In the case of fingerprint recognition however, only an average of less than 40 minutiae are extractable from a fingerprint, and when the pool cardinality (which is equivalent to number of

minutiae in the fingerprint) is too small, mean-pooling produces a better foreground-background separation [225].

On the other hand, the performances of *SQ+SUMPOOL* and *SQ+MEANPOOL* are not as similar to each other as observed between *HQ+SUMPOOL* and *HQ+MEANPOOL*. This is because Euclidean distance reacts to averaging differently – histogram intersection measures the similarity of the shapes of two histograms, whereas Euclidean distance measures the numerical difference between two vectors in the ℓ^2 sense. Besides, vectors generated by soft quantization consist of more non-zero values and so, are more sensitive to the dividing factor in mean-pooling function.

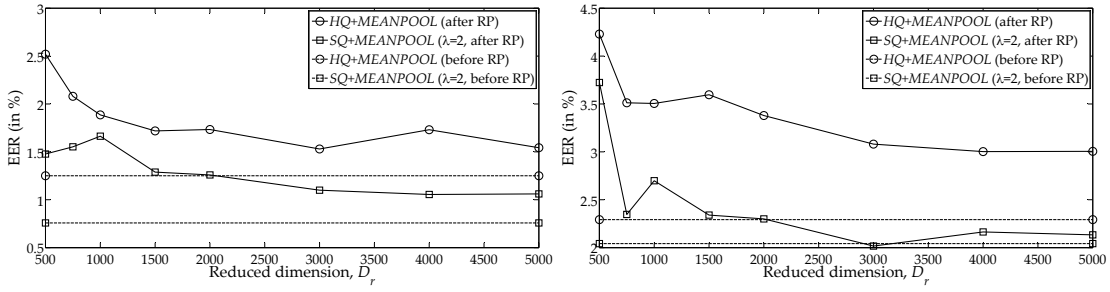
Moreover, as the dictionary size (K) increases, the EER for all settings drop. Larger dictionary size allows a finer quantization of the minutia vectors and reduces the quantization error, hence improves the recognition rate of fingerprints. The EERs reach their lowest points at $K = 5000$ and saturate or even begin to increase thereafter due to redundancy.

It is also notable that soft quantization with larger target sparsity (λ) does not perform as good as smaller λ when K is small. The target sparsity controls the fuzziness of soft quantization in dictionary learning. Even though such property helps to improve the performance of vector quantization by allowing one minutia vector to be assigned to multiple atoms, exceedingly high sparsity results in over-fuzziness and causes the minutia vectors to lose their uniqueness.

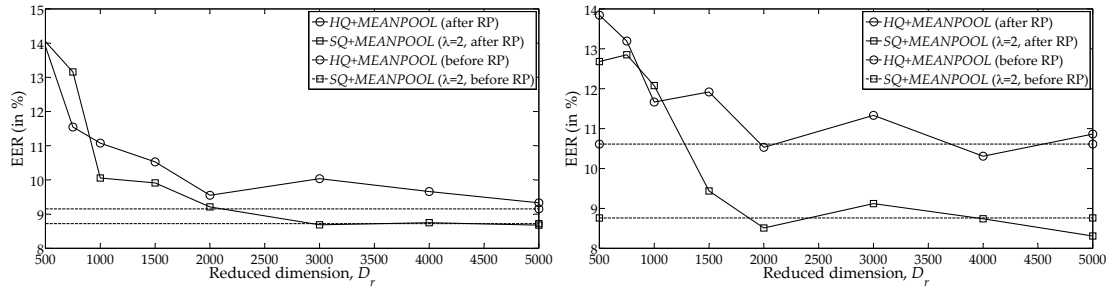
Comparing the two minutiae labelling options, soft quantization performs better than hard quantization. For soft quantization, the partition borders are fuzzy to allow multiple assignments of one minutia vector, resulting in lower quantization error. Besides, there are numerous effective optimization algorithms to solve soft quantization problem that provide high fidelity.

5.5.3 Verification Rate of Cancellable Fixed-Length Representation

The experiments in this section inherit the optimal parameter values observed in the section 5.5.2, i.e. $K = 5000$ and $\lambda = 2$ (for soft quantization only). In addition, mean-pooling function is chosen. Figure 5.3 shows the stolen-key performance of the proposed cancellable fixed-length representation when the vector dimension after RP (D_r) is reduced to the range between 500 and 5000.



(a) FVC2002 DB1. **(b)** FVC2002 DB2.



(c) FVC2004 DB1. **(d)** FVC2004 DB2.

Figure 5.3: Performance of the proposed minutiae-based fingerprint template after S2V transformation via BoM with different parameter values. The results correspond to FVC2002 DB1 dataset.

For some datasets such as FVC2004 DB1 and FVC2004 DB2, the EERs after RP is able to match with the original EERs with certain D_r values, while performance drop is observed in other datasets even when $D_r = K$. This confirms the discussion in section 4.6.3 that the data distribution of the original vector is one of the factors affecting the performance after RP. Just like MLC, the BoM-generated vector is sparse. For example, assuming that there are 40 minutiae in the input fingerprint, the highest possible cardinality of the vector generated is 40 (for hard quantization) or 80 (for soft quantization with $\lambda = 2$) out of 5000 elements ($K = 5000$). Therefore, RP does not adapt to the BoM-based transformation as good as the KPCA-based transformation.

Despite that, a similar trend in the EERs can be observed as the dimension is reduced. The EERs increase gradually but marginally within the range of $2000 \leq D_r \leq 5000$ and begin to rocket when $D_r < 2000$. The EERs at $D_r = 2000$ are 1.73%, 3.38%, 9.55% and 10.53% for hard quantization, and 1.26%, 2.30%, 9.21% and 8.51% for soft quantization when tested on the four fingerprint datasets respectively. These EER values are reflected in Table 5.2, together with the past results obtained in this thesis as well as other existing methods. The proposed method is able to preserve information in the MLC template as the performance approximate to the performance of the original

MLC. Although the EERs of the BoM-based method are lower than the benchmarking methods in most cases, the KPCA-based method still has the lowest EER among all.

Table 5.2: Summary of the recognition accuracy (in terms of EER) of the proposed fixed-length cancellable fingerprint template compared to other existing methods.

Algorithm	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
Proposed method				
Original MLC (before S2V transformation, Chapter 3)	2.83	2.25	9.16	8.89
HQ+MEANPOOL (after S2V transformation)	1.73	3.38	9.55	10.53
SQ+MEANPOOL (after S2V transformation)	1.26	2.30	9.21	8.51
Past results and other existing S2V transformation methods (both cancellable and non-cancellable)				
KPCA-based method[1] (Chapter 4)	**0.51**	2.00	**4.29**	**5.51**
Nagar et al.[1] [108]	-	3.00	-	-
Bringer and Despiegel[2] [47]	-	1.70	-	-
Vij and Namboodiri[2] [197]	1-2	**1-2**	7-8	8-9
Nandakumar[2] [137]	0.80	0.70	-	-

[1]cancellable method.

[2]non-cancellable method.

5.5.4 Security and Privacy Analyses

For the sake of comparability, the same set of apparatus used in Chapter 3 and 4 are adopted to evaluate the security and privacy of the cancellable fingerprint template generated through the BoM model. The analyses are done based on $K = 5000$, $\lambda = 2$ (for soft quantization only), $D_r = 2000$ and with mean-pooling function applied on both minutiae labelling techniques.

Non-invertibility: Resistance against Reverse Attack

The non-invertibility of the proposed scheme is analysed with the assumption that the protected template and all helper data are compromised. In this case, the helper data includes the random matrix used for RP (**R**) and the dictionary obtained from the training stage of BoM modelling (**C**).

Table 5.3: Ratio between the average cardinality values of the proposed vector before and after RP, $\frac{\|\mathbf{V}_{\text{BoM}}\|_0}{\|\hat{\mathbf{V}}_{\text{BoM}}\|_0}$.

Dataset	*HQ+MEANPOOL*	*SQ+MEAPOOL*
FVC2002 DB1	0.02	0.03
FVC2002 DB2	0.02	0.04
FVC2004 DB1	0.02	0.04
FVC2004 DB2	0.02	0.03

As discussed in section 5.5.3 that the BoM-generated vector is sparse and thus, the RP operation may be exposed to the risk of reverse attack (refer to section 3.6.4 for further explanation). According to Figure 5.3, the cardinality ratio of the RP transformation is much lower than 0.13, the possibility is very high that the original vector can be reconstructed from the cancellable template. However, there are two more phases protecting the raw minutiae set.

In BoM, the dictionary, is the helper data obtained from the training stage and is stored for template generation upon authentication. The dictionary consists of the prototypes of minutiae vectors, also known as the centroids (in hard quantization) or atoms (in soft quantization). Comparing the two BoM models, hard quantization takes less effort to crack than soft quantization. Given a histogram generated by hard quantization, it leaks the information about the total number of minutiae in the fingerprint and the original minutia vectors can be estimated by matching the histogram with the dictionary. On the contrary, the adversary cannot relate a soft quantization-generated vector to the number of minutiae in each partition regardless of the pooling function used as it contains only the pooled weight of the minutiae assignment. Besides, the dictionary size is another factor affecting the privacy of the scheme. Larger K leads to finer space partitioning and thus, the original minutia vectors are easier to crack. However, as seen in Fig. 5.2, the performance drops when K is too small. There exists a trade-off between performance and privacy in the choice of K.

Lastly, the non-invertibility of the MLC algorithm has been presented in section 3.6.4. One of the advantages of the proposed multi-phase cancellable fingerprint template generation scheme is that it provides multi-layer protection to the raw fingerprint data.

Unlinkability: Resistance against Linkage Attack

The unlinkability of the proposed cancellable template is measured by the separability between templates generated using multiple distinct random keys as explained in

Table 5.4: Separability of the proposed cancellable fixed-length representation expressed in the form of "separability$(\mu_{SKG}, \sigma^2_{SKG})[\mu_{DKG}, \sigma^2_{DKG}]$". μ_{SKG} and σ^2_{SKG} represent the mean and variance of the same-key genuine matching distribution, while μ_{DKG} and σ^2_{DKG} are the equivalent parameters of the different-key genuine matching distribution. Since the decimal values shown are rounded to the nearest 0.01, any value that is less than 0.005 are written as <0.005.

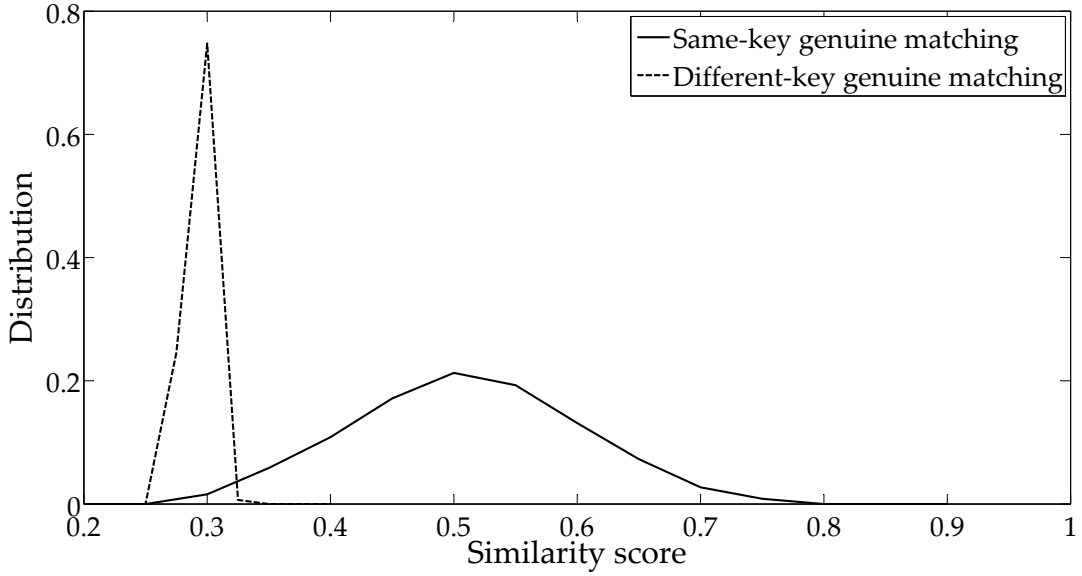
Algorithm	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
HQ+MEANPOOL	2.99 (0.45,0.01) [0.29,<0.005]	3.06 (0.47,0.01) [0.29,<0.005]	2.01 (0.39,<0.005) [0.29,<0.005]	2.24 (0.40,<0.005) [0.29,<0.005]
SQ+MEANPOOL	3.03 (0.49,0.01) [0.29,<0.005]	3.38 (0.51,0.01) [0.29,<0.005]	2.24 (0.41,0.01) [0.29,<0.005]	2.30 (0.43,0.01) [0.29,<0.005]

section4.6.4. Table 5.4 shows the information of the same-key and different-key distributions and the resulting separability values. One can observe that the means are at least 0.1 apart from each other and the variance of both distributions are very small ($\leq$ 0.01). By comparing the settings with the highest separability (*SQ+MEANPOOL*, FVC2002 DB2) and the lowest separability (*HQ+MEANPOOL*, FVC2004 DB1) as visualized in Figure 5.4, it is obvious that the intersected area between the same-key and the different-key distributions is less in the former case.

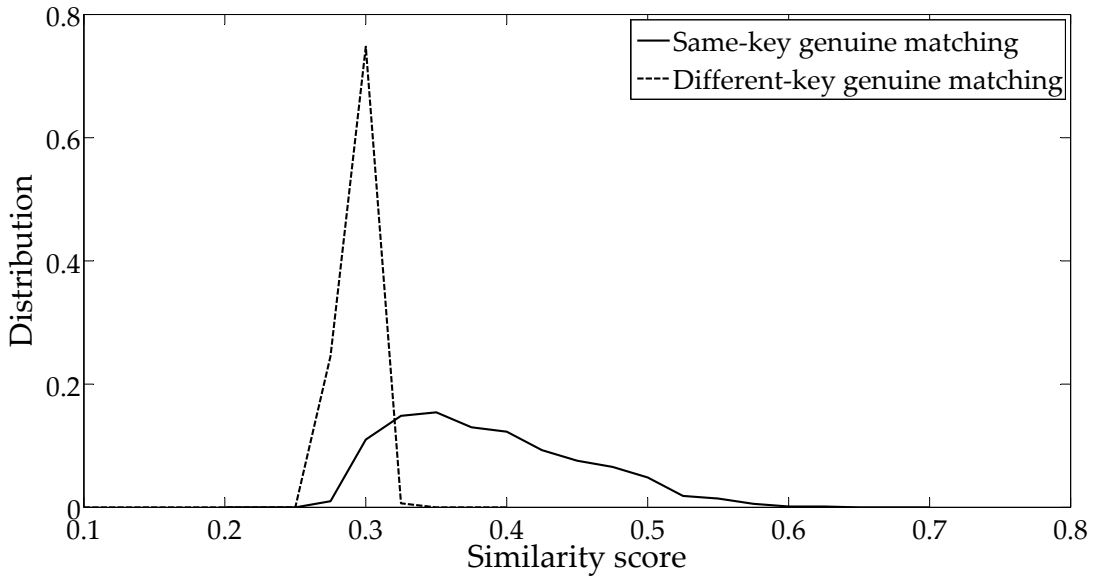
Furthermore, the BoM-based cancellable template yields lower separability than the KPCA-based method (in Table 4.3). The analysis in section 5.5.4 is based on the template dimension, $D_r = 60$, and $D_r = 2000$ is used in this section. It is rational that the one with longer vector length should have better unlinkability as it is able to accommodate more combinations of the vector values. However, since the vector generated by BoM modelling is sparse, the values of the final cancellable template are also limited.

Entropy: Resistance against Brute Force Attack

The discrete entropy and differential entropy of the proposed cancellable template is computed based on (3.6.6) and (3.6.8) respectively. The entropies of the cancellable template are similar to each other regardless of the minutiae labelling technique. Even though soft quantization produces vectors with higher cardinality, zero is still the dominant element value in the vectors. Therefore, the entropies of the cancellable templates after RP do not differ much between the two techniques. The averaged total discrete entropies over all datasets are 5105.36 bits and 5107.59 bits for *HQ+MEANPOOL* and



(a) *SQ+MEANPOOL*, FVC2002 DB2 (highest separability).



(b) *HQ+MEANPOOL*, FVC2004 DB1 (lowest separability).

Figure 5.4: Examples of same-key and different-key distributions.

SQ+MEANPOOL respectively, and the discrete entropy per component is 2.55 bits for both techniques.

Despite the total entropy of the BoM-based method being a lot higher than that of the KPCA method (refer to Table 4.4) due to longer vector length, the average entropy per component is halved compared to the KPCA method. This is caused by the sparseness of the vector before RP as discussed in section 4.6.4.

Table 5.5: Entropy (in bits) of the proposed cancellable fixed-length representation of fingerprint. The first number represents the average discrete entropy per vector component and the second number represents the total discrete entropy of the vector. The number in parenthesis is the average differential entropy per component.

Algorithm	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
HQ+MEANPOOL	2.55, 5109.49 (-4.11)	2.55, 5102.58 (-4.11)	2.55, 5098.36 (-4.12)	2.56, 5111.00 (-4.11)
SQ+MEANPOOL	2.56, 5110.32 (-4.11)	2.55, 5098.69 (-4.11)	2.55, 5109.38 (-4.11)	2.56, 5111.97 (-4.11)

5.5.5 Computational Complexity

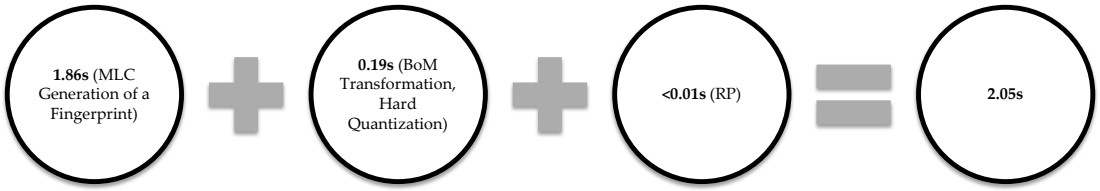
If hard quantization is chosen as the BoM transformation option, the time complexity of the training stage is equivalent to the complexity of Llyod's algorithm, which is $\mathcal{O}(KN_tD_mI)$. The transformation stage of hard quantization involves the K-nearest neighbour search operation. A simple linear search approach yields a time complexity of $\mathcal{O}(KN_mD_m)$. As for soft quantization, the K-SVD algorithm is computationally bounded by $\mathcal{O}(KN_tD_m^2I)$. At the transformation stage, only OMP is performed and it has a complexity of $\mathcal{O}(N_m\lambda)$ (refer to Algorithm 5.1). Considering the entire training and template generation process which include MLC generation ($\mathcal{O}(N_mD_m)$ per fingerprint) and RP ($\mathcal{O}(KD_r)$), the time complexities of the two stages of a fingerprint authentication system in respective order are $\mathcal{O}(\max(KN_tD_mI, N_tN_mD_m))$ and $\mathcal{O}(KN_mD_m)$ for hard quantization, and $\mathcal{O}(\max(KN_tD_m^2I, N_tN_mD_m))$ and $\mathcal{O}(KN_mD_mI)$ for soft quantization.

The CPU runtime of the proposed scheme is shown in Table 5.6. Considering the template generation stage, *HQ+MEANPOOL* requires slightly more time than *SQ+ MEANPOOL*. The linear nearest-neighbour search algorithm performs distance calculation between the minutia vectors and the centroids in the dictionary; whereas the OMP algorithm merely involves inner product operations and other linear algebraic operations for λ times per minutia vector. Therefore, the former consumes more time, but only by a little. From the CPU runtime breakdown charts in Figure 5.5 and 5.6, hard quantization and soft quantization for FVC2004 DB2 take 0.19s and <0.01s to complete respectively, making the BoM method an advantage over the KPCA-based method (refer to Figure 4.9) in terms of template generation speed. Also note that the MLC generation phase requires less time than that in Figure 4.9 as a different MLC parameter set is used.

Table 5.6: CPU runtime of the proposed cancellable fingerprint template generation scheme, running on MATLAB environment (Windows 7) with an Intel® Core™ i5-2430M 2.40GHz processor. The number of iterations, $I = 50$ is used.

Stage	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
HQ+MEANPOOL				
Training	1216.88s	1655.82s	1769.55s	1743.08s
Template Generation	1.36s	1.82s	2.04s	2.05s
SQ+MEANPOOL				
Training	3336.43s	3856.58s	3993.17s	3901.77s
Template Generation	1.17s	1.60s	1.83s	1.86s

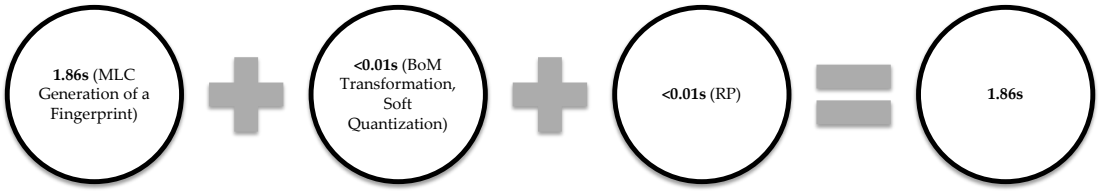
(a) Training stage.



(b) Template generation stage.

Figure 5.5: CPU runtime breakdown chart of training stage and template generation stage for FVC2004 DB2 using hard quantization.

(a) Training stage.



(b) Template generation stage.

Figure 5.6: CPU runtime breakdown chart of training stage and template generation stage for FVC2004 DB2 using soft quantization.

On the other hand, *SQ+MEANPOOL* takes more than double the CPU runtime of *HQ+MEANPOOL* during training. The training stage of BoM involves two steps, namely minutia vectors assignment and dictionary update, while template generation only requires the assignment step. Since it has been discussed above that the assignment step of hard quantization is more time-consuming than soft quantization, it is clear that the dictionary update step of soft quantization consumes much more time than hard quantization. Soft quantization updates the dictionary through K SVD operations while hard quantization merely re-calculates the centroids of the clusters. As a result, the training stage for soft quantization has higher CPU runtime than both hard quantization and the previously proposed KPCA-based S2V transformation.

5.6 Summary

A S2V transformation technique via BoM modelling has been proposed in this chapter. The concept of BoM modelling, which is originated from BoW modelling, is consolidated from the aspect of three design approaches, viz. minutiae representation, space partitioning and minutiae labelling. Two options from the minutiae labelling aspect are demonstrated throughout the chapter, namely hard quantization via K-means clustering and soft quantization via dictionary learning. Since both minutiae labelling techniques are using *a posteriori* space partitioning, the training for dictionary is necessary prior template generation.

The experimental procedures follow the ones in the previous chapters (Chapter 3 and 4). As shown in Table 5.2, comparing the two minutiae labelling techniques, soft quantization outperforms hard quantization due to its fuzziness in minutia vectors assignment. Also, the performance obtained for the BoM-based method approximate to the performance of the original MLC, but is not as good as the KPCA-based method introduced in Chapter 4.

From the aspect of aspect of security and privacy, the original fixed-length vector can be easily revealed from the cancellable template as it is sparse. In spite of this privacy weakness, the raw minutiae set is protected by the MLC algorithm and the proposed BoM transformation itself. For this, soft quantization has stronger non-invertibility than hard quantization. The inter-templates separability and entropy per vector component are also weaker than the KPCA-based method. Overall, the BoM-based can-

cellable fingerprint template is less robust in terms of non-invertibility, unlinkability and entropy compared to the KPCA-based method, but still in acceptable range.

As discussed before that since the training stage is performed off-line, the time taken for training, albeit seemingly long, is omissible. Looking at the template generation time, the BoM modelling, especially the soft quantization technique has a great advantage over other methods.

In a nutshell, soft quantization outperforms hard quantization in all aspects investigated, including recognition accuracy, security, privacy, and computational complexity. Although the BoM-based method does not perform as good as the KPCA-based method, it is a tangible solution towards the application of variable-size and unordered minutiae-based templates in various classifiers and bio-cryptosystems, biometric template binarization, and direct vector-to-vector comparison.

CHAPTER 6

Cancellable Fingerprint Bit-String Generation

6.1 Introduction

In this chapter, fixed-length cancellable fingerprint templates proposed previously, including the KPCA-based template (Chapter 4) and BoM-based template (Chapter 5), are converted into binary form. The template binarization phase is appended to the existing schemes as depicted in Figure 6.1. In here, bit-string refers to fixed-length binary vector.

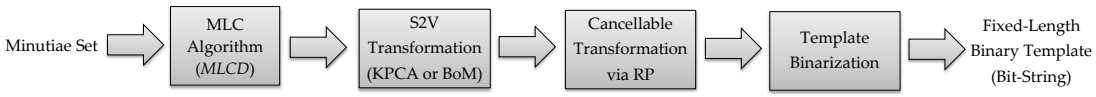


Figure 6.1: Block diagram of the proposed bit-string generation scheme.

There are several benefits of storing biometric templates in binary form. First of all, most applications such as bio-cryptosystems and fingerprint indexing requires the input to be bit-strings. Comparison between two binary templates merely involve logical operations rather than real-valued arithmetic operations, hence improves the speed of fingerprint matching. Also, binary templates consume less storage than real-valued templates.

6.2 Nomenclature

Symbol	Description
$\hat{\mathbf{V}}_{\text{BoM}} \in \mathbb{R}^{D_r}$	cancellable fixed-length fingerprint template generated through BoM modelling
$\hat{\mathbf{V}}_{\text{KPCA}} \in \mathbb{R}^{D_r}$	cancellable fixed-length fingerprint template generated through KPCA
$\hat{\mathbf{V}}_b \in \{0,1\}^{D_b}$	cancellable fingerprint bit-string
D_r	dimension of the cancellable template before binarization
D_b	final bit-length — dimension of the cancellable bit-string
$\hat{\mathbf{V}}_{\text{train}} \in \mathbb{R}^{D_r \times N_t}$	training samples for DQ, can be either KPCA- or BoM-generated feature vector
N_t	total number of training samples for DQ
N_{spu}	number of training samples per user
SNR$(\cdot)$	signal-to-noise ratio of a feature component
σ_u^2	user's variance
σ_g^2	global variance
N_d	maximum number of bits assigned to a feature component for DQ
$\Gamma_{(ij)}$	reliability of the jth bit position in the ith vector component
$\mathbf{Q}$	user-specific quantization helper data, including the number of bits and quantization intervals for each vector component

6.3 Binarization Methods Used for the Proposed Fingerprint Bit-String Generation Scheme

The quantization techniques used for biometric template binarization can be categorized into static quantization and dynamic quantization. Static quantization assigns equal number of bits to each data component, whereas dynamic quantization allows

more bits to be assigned to more discriminative vector components in order to minimize the false acceptance rate (FAR) or/and false rejection rate (FRR) of the system.

The most popular static quantization technique used in biometric template binarization is the single-bit or multi-bit fixed threshold quantization method [47, 100, 103, 107, 108, 110, 123, 137, 196, 197], Fuksis et al. [226] demonstrated a bit-stability-based error computation scheme. This scheme extends the BioHash algorithm [110] by weighting the error rate contribution of each bit according to its stability. Vielhauer et al. [227] defined the genuine quantization interval for each feature of a user as $[x_{\min}(1-t), x_{\max}(1+t)]$, where $x_{\min}$ and $x_{\max}$ are the lowest and the highest recorded value of the feature respectively and t is a tolerance factor. The remaining intervals are constructed with the same width as the genuine interval.

The binarization schemes listed above are static approaches. As an example of dynamic approach, Feng and Wah [228] and Chang et al. [229] employed a multi-state discretization (MsD) method, of which the genuine boundary of each feature component is defined as $[\mu - k\sigma, \mu + k\sigma]$, where μ and σ denote the mean and the standard deviation of the user distribution respectively. The boundary is divided into several segments according to a predefined value for each feature. Later, Teoh et al. [112, 230] proposed a user-dependent MsD method which converts the real feature space into the index space followed by Gray coding. Derived from the reliability-based static bit selection method [158], Lim et al. [231] introduced a dynamic bit allocation scheme. Unlike its predecessor, the number of bits allocated to a feature dimension is determined dynamically according to the bit-stability as well as the signal-to-noise ratio (SNR) of the feature component. Other dynamic approaches include mutli-bit detection rate optimized bit allocation (DROBA) [232] and the area under the FRR curve optimized bit allocation (AUF-OBA) principle [233].

The template binarization methods used for the proposed scheme are described below.

6.3.1 Static Quantization: Zero-Thresholding

As discussed above, fixed threshold quantization is most commonly used among the various static quantization techniques mainly because of its simplicity. In this chapter, the same technique is used by setting the fixed threshold to zero so that $\hat{V}_{b(i)} = 1$ when $\hat{V}_{(i)} > 0$ and $\hat{V}_{b(i)} = 0$ otherwise, where $\hat{V}_{(i)}$ and $\hat{V}_{b(i)}$ are the ith component of the real-valued vector (may it be $\hat{\mathbf{V}}_{\text{KPCA}}$ or $\hat{\mathbf{V}}_{\text{BoM}}$) and the bit-string ($\hat{\mathbf{V}}_b \in \{0,1\}^{D_b}$) respectively.

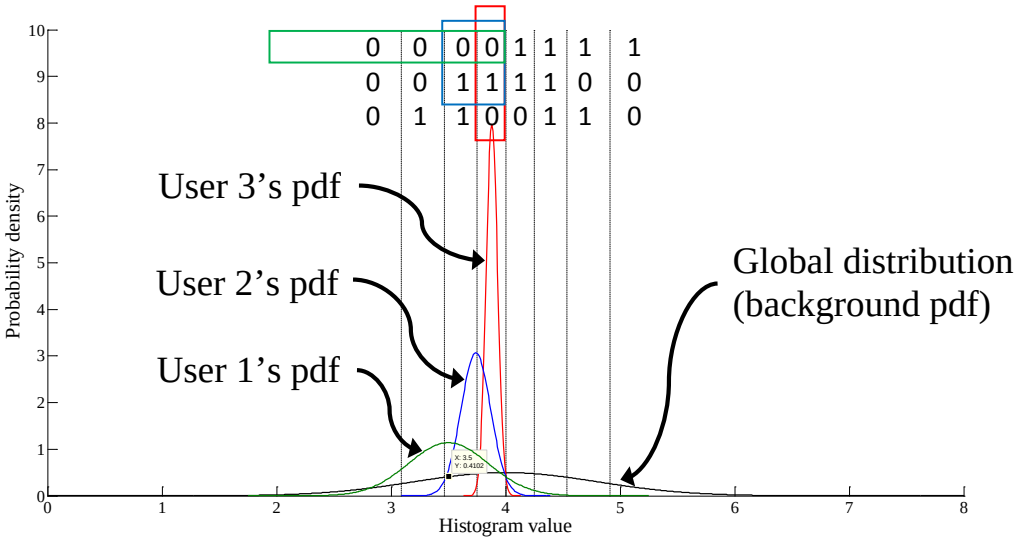


Figure 6.2: DQ technique adopted for the proposed fingerprint bit-string generation scheme. The background pdf (*black*) is first quantized in equal-probable manner with $N_d = 3$ bits. User 1 (*green*) has lower discriminability due to high intra-user variance, hence is only assigned with one bit ('0'); user 2 (*blue*) yields lower intra-user variance and the data distribution is concentrated in the range of '011' and '010', so it is assigned with two bits ('01'); lastly, user 3 is the most discriminative among the three users and is assigned with fully three bits ('010').

Since this is a single-bit quantization method, the final bit-length is equivalent to the length of the original vector, i.e. $D_b = D_r$.

6.3.2 Dynamic Quantization

In this chapter, the dynamic quantization (DQ) technique presented by Lim et al. [231] is adopted. Figure 6.2 provides a pictorial illustration of the overall concept of DQ. The original method assumes the input data to be normally distributed. Regarding this, although RP was introduced as a cancellable transformation method, it also has the ability to normalize the original vector, regardless of the original distribution. Therefore, even the sparse vector generated by BoM is normally distributed in its cancellable form. This is important as the pdf of the input data is the main factor determining the initial quantization intervals of DQ. If the true data distribution disagrees with the estimated distribution, over-populated and redundant intervals may exist, and the discriminability of the vector components cannot be accurately derived.

DQ is a training-based vector binarization technique, so it can be divided into the training stage and the bit-string generation stage. Let $\hat{\mathbf{V}}_{\text{train}} \in \mathbb{R}^{D_r \times N_t}$ be the training set, the

data component- and user-specific quantization information for each user is trained by following the steps below:

1. Statistical analysis: the discriminability of a feature component is defined as its signal-to-noise ratio, which is derived from the statistical measurement

$$\mathrm{SNR}(\hat{V}_{\mathrm{train(i)}}) = \frac{\sigma^2_{g(i)}}{\sigma^2_{u(i)}}, \tag{6.3.1}$$

   where $\hat{V}_{\mathrm{train(i)}}$ is the ith component of the user's feature vector and $\sigma^2_{g(i)}$ and $\sigma^2_{u(i)}$ are the inter-user variance and intra-user variance respectively. The components are then sorted according to the discriminability value in descending order.

2. Background pdf quantization: the background pdf refers to the aggregated data distribution of all users. Each feature component is initially converted into a N_d-bit binary string, $\hat{\mathbf{V}}_{\mathrm{train},d} \in \{0,1\}^{N_d}$, according to equal-probable quantization based on the background pdf as shown in Fig. 6.2.

3. Reliability weight computation: in this step, the reliability of every bit position is evaluated by counting the number of occurrences of 1. The reliability of the jth bit position in the ith vector component is formulated as

$$\Gamma_{(ij)} = \frac{1}{N_{spu}} \sum_{k=1}^{N_{spu}} V_{\mathrm{train},d(ijk)} \tag{6.3.2}$$

   where $V_{\mathrm{train},d(ijk)}$ denotes the jth bit in the initially binarized ith vector component of the kth training sample and N_{spu} is the number of training samples per user. The reliability measure is then re-scaled so that it addresses both agreeing 1's and 0's, depending on the majority:

$$\Gamma_{(ij)} = \begin{cases} \Gamma_{(ij)}, & \text{if } \Gamma_{(ij)} \geq 0.5; \\ 1 - \Gamma_{(ij)}, & \text{otherwise.} \end{cases} \tag{6.3.3}$$

4. Discriminability- and reliability-based bit allocation: re-visit the feature components sorted in step 1 and choose the bits with reliability higher than a pre-defined threshold value. The process ends when the total number of bits allocated reaches the desired bit-length, D_b.

The quantization intervals and the number of bits assigned for each component are stored as the user-specific quantization information, **Q** required for bit-string generation upon fingerprint authentication. The final cancellable bit-string is represented as $\hat{\mathbf{V}}_b \in \{0,1\}^{D_b}$.

6.4 Experiments and Analyses

6.4.1 Testing Protocol

The experimental protocol follows those described in the previous chapters (Chapter 4 and 5). Six out of eight samples per fingerprint are used for training while the remaining samples are used for matching. The procedure is repeated five times, each with different randomly chosen samples, to obtain average results. The bit-strings are matched with one minus normalized Hamming distance:

$$S = 1 - \frac{\|\hat{\mathbf{V}}'_b \oplus \hat{\mathbf{V}}''_b\|_1}{D_b}, \tag{6.4.1}$$

where $\hat{\mathbf{V}}'_b$ and $\hat{\mathbf{V}}''_b$ are two instances of the proposed cancellable bit-string.

Table 6.2 recapitulates the important parameters used for different S2V transformation methods as suggested in Chapter 4 and 5.

6.4.2 Verification Rate of Cancellable Bit-String

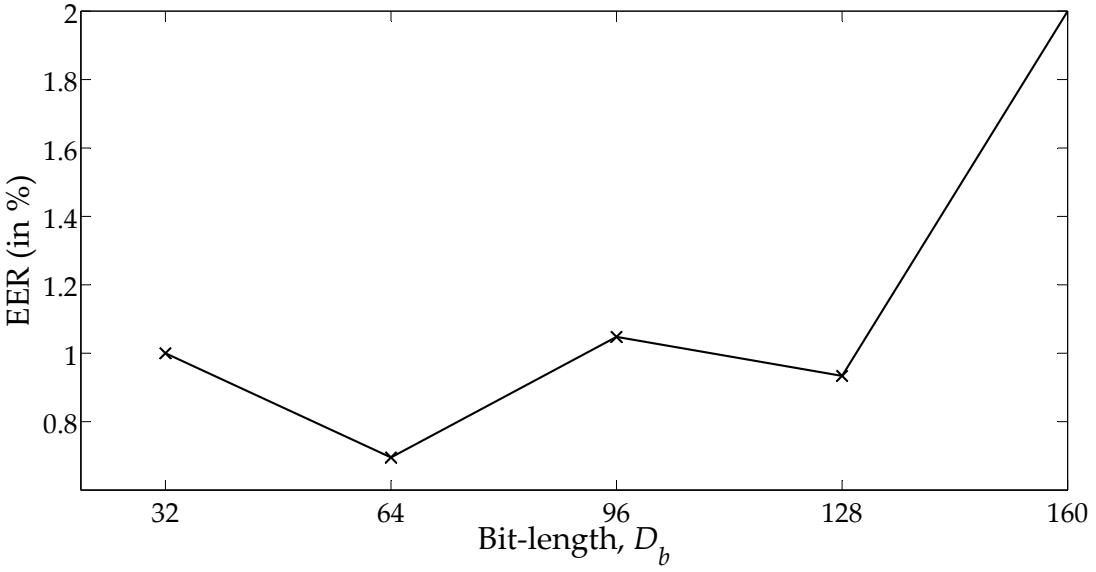
In this experiment, for DQ, the vector space is initially quantized into a 3-bit binary space, i.e. $N_d = 3$. Figure 6.3 shows the EERs of the final bit-strings generated by DQ while varying the bit-length (D_b). Regardless of the S2V transformation method used, the EER plot follows similar trend — as D_b increases, the EER decreases until it reaches its minimum and beats off afterwards. DQ selects bits based on feature discriminability and bit stability. If D_b is too small, there would be insufficient number of bits to address inter-class variability. On the other hand, when D_b continues to increase beyond the minimum point of EER, even feature components with little discriminability will be selected and reduce the overall inter-class dissimilarity of the final bit-string. From Figure 6.3, the bit-lengths with optimal performance for the KPCA-based method, *HQ+MEANPOOL* and *SQ+MEANPOOL* are 64 bits (out of $D_r \times N_d = 60 \times 3 = 180$ bits), 768 bits (out of 6000 bits) and 1024 bits (out of 6000 bits)

Table 6.2: Summary of the parameters used for the S2V transformation methods.

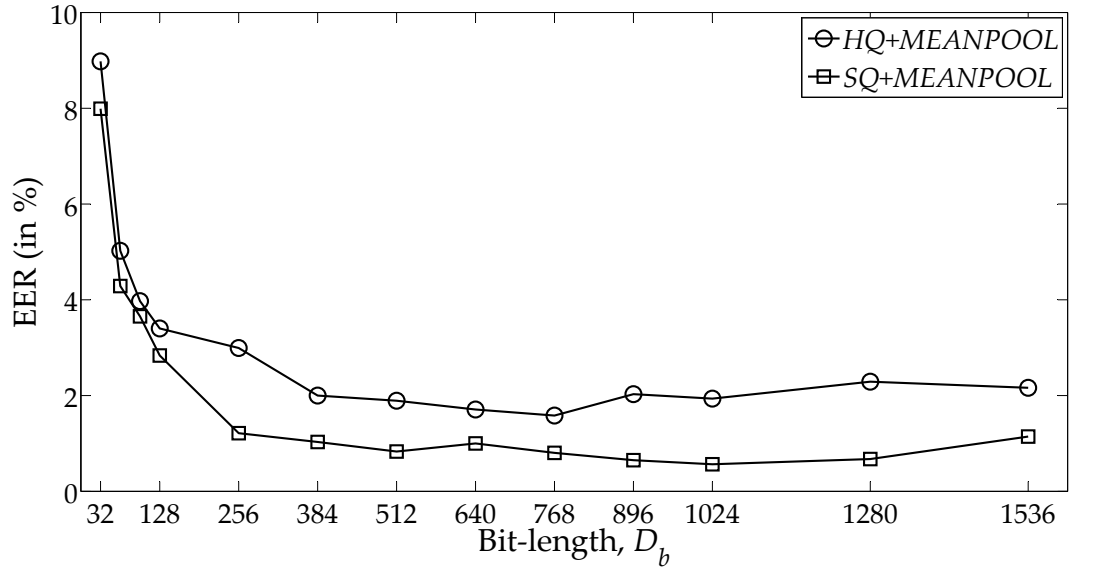
Parameter	Description	KPCA-based method	*HQ+ MEANPOOL*	*SQ+ MEANPOOL*
N_t	total number of training samples	600	600	600
N_p	number of principal components chosen for KPCA, also the dimension of the KPCA-based fixed-length vector	125	-	-
K	dictionary size for BoM modelling, also the dimension of the BoM-based fixed-length vector	-	5000	5000
D_r	dimension of the final cancellable template	60	2000	2000

respectively. Although *SQ+MEANPOOL* may have the longest bit-length, the KPCA-based method has the highest optimal bit-length versus maximum possible bit-length ratio ($\frac{64}{180} \approx 0.36$). It implies that the KPCA-generated vector contains more discriminative components than the other two methods. This agrees with the observation while comparing the performance among the three methods in both real and binary form as in Table 6.3.

Comparing the two template binarization methods, it is obvious that DQ performs better than static quantization, or specifically, zero-thresholding. While zero-thresholding is barely equivalent to applying a signum function on every vector component, DQ systematically allocates bits to individual component according to its discriminability. The discriminability is measured by the signal-to-noise ratio as defined in (6.3.1), which is fundamentally the ratio between the inter-class variance and intra-class variance. Therefore, DQ performs bits allocation in such a way that the inter-class dissimilarity and intra-class similarity are maximized, resulting in better recognition accuracy of the final bit-string.



(a) KPCA-based method.



(b) BoM-based method.

Figure 6.3: Performance of the cancellable bit-string generated by DQ while varying the bit-length.

Moreover, not only is zero-thresholding more inferior than DQ, performance deterioration is observed when zero-thresholding is applied compared to the real-valued template. On the other hand, DQ promotes performance-lifting effect after binarization. This can be verified by observing the genuine-impostor distributions in Figure 6.4. When zero-thresholding is used, although the means of both genuine and impostor distributions increase, the overlapping area between them also increase, leading to higher error while making accept/reject decision. For DQ however, the distributions

Table 6.3: Recognition accuracy (in terms of EER) of the proposed cancellable bit-string compared to other existing methods. In the table, *ZT* represents zero-thresholding while *DQ* represents dynamic quantization.

Algorithm	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
Proposed methods after binarization				
Original MLC (Chapter 3)	2.83	2.25	9.16	8.89
KPCA+ZT	2.59	3.24	7.80	7.88
HQ+MEANPOOL+ZT	2.42	2.23	9.81	9.02
SQ+MEANPOOL+ZT	0.93	0.93	8.40	8.00
KPCA+DQ	0.70	**0.36**	**4.09**	**5.03**
HQ+MEANPOOL+DQ	1.58	1.61	8.50	7.52
SQ+MEANPOOL+DQ	0.57	0.56	8.18	7.38
Past results and other existing fingerprint bit-strings (both cancellable and non-cancellable)				
KPCA-based method[1] (Chapter 4)	**0.51**	2.00	4.29	5.51
HQ+MEANPOOL[1] (Chapter 5)	1.73	3.38	9.55	10.53
SQ+MEANPOOL[1] (Chapter 5)	1.26	2.30	9.21	8.51
Nagar et al.[1] [108]	-	3.00	-	-
Bringer and Despiegel[2] [47]	-	1.70	-	-
Vij and Nambood-iri[2] [197]	1-2	**1-2**	7-8	8-9
Nandakumar[2] [137]	0.80	0.70	-	-

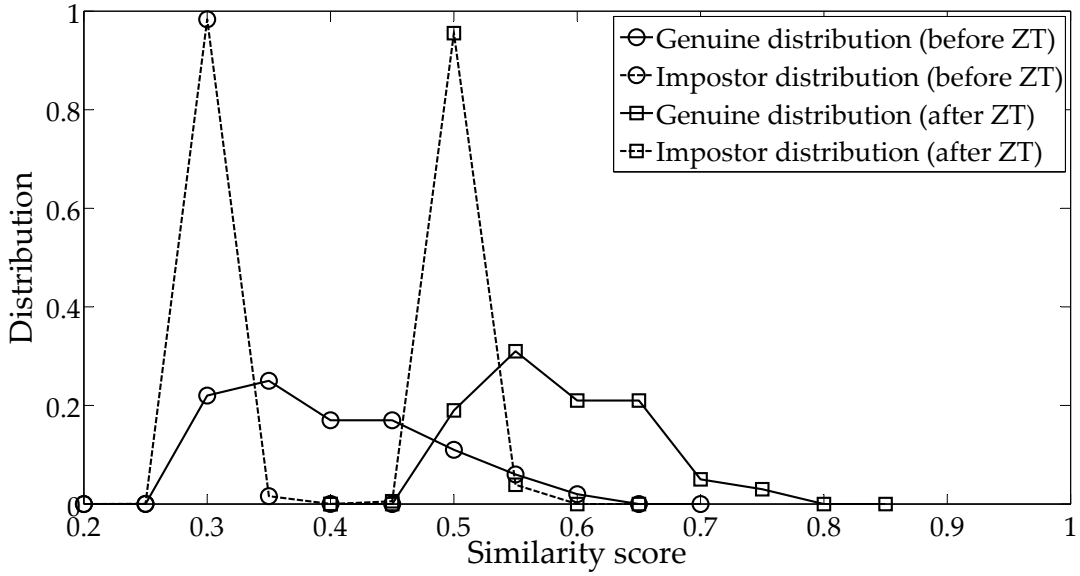
[1]cancellable method.

[2]non-cancellable method.

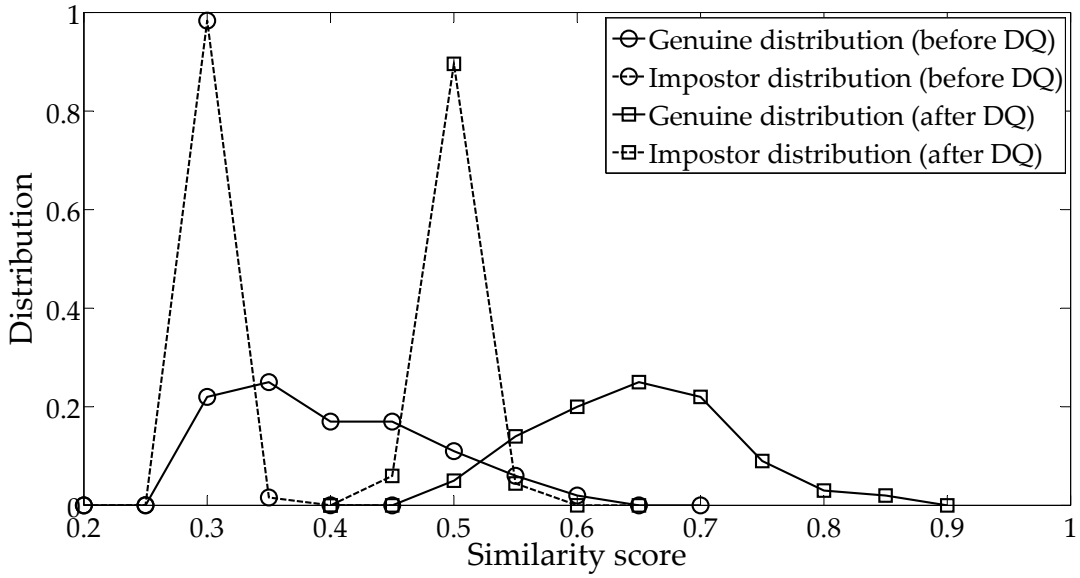
are further to each other after binarization, thus reducing the probability of false recognition.

6.4.3 Security and Privacy Analyses

The security and privacy of the MLC algorithm and the S2V transformations have been presented in Chapter 3, 4 and 5. In this section, the analyses are primarily based on the binarization methods with the optimal bit-lengths (D_b) of 60 bits, 2000 bits, 2000 bits, 64 bits, 768 bits and 1024 bits for *KPCA+ZT*, *HQ+MEANPOOL+ZT*, *SQ+MEANPOOL+ZT*, *KPCA+DQ*, *HQ+MEANPOOL+DQ* and *SQ+MEANPOOL+DQ* respectively.



(a) Zero-thresholding (ZT).



(b) Dynamic quantization (DQ).

Figure 6.4: Examples of genuine-impostor distributions when soft quantization of the BoM model is used for S2V transformation before binarization (*SQ+MEANPOOL*) and after binarization (*SQ+MEANPOOL+ZT* or *SQ+MEANPOOL+DQ*). Figure shows the results on FVC2004 DB2.

Non-invertibility: Resistance against Reverse Attack

If zero-thresholding is chosen as the binarization method, the compromised bit-string reveals the sign of the real values in the original template before binarization. The improbability of reversing the bit-string depends on the range of the real values (which also determines the quantization width). For example, the proposed KPCA-based can-

cellable real template has values ranging from -0.4 to 0.4 while the BoM-based template ranges from -0.1 to 0.1, hence the former has stronger non-invertibility than the latter. However, this piece of information is kept secret from the adversary even if he has the knowledge about the template generation method as only the bit-string is stored in the database for authentication.

For DQ, the helper data stored includes the quantization intervals and the number of bits assigned to each vector component. If this information, together with the bit-string, are compromised, the adversary is able to know the range of the real vector values. Consequently, the original real vector may be estimated based on the information. The accuracy of the estimation depends on the quantization widths.

Comparing the two binarization methods, DQ yields weaker non-invertibility than zero-thresholding in general. Zero-thresholding has fixed number of bit(s) and quantization width for every vector component. For DQ, the non-invertibility strength of individual vector component is seemingly the same as single-bit or multi-bit fixed-threshold quantization, except that the quantization widths are known to the adversary. Therefore, DQ provides more information regarding the original real vector. In special case where no bit is assigned to a vector component, no information is stored regarding that component and so, its value is kept secret. This situation can be noticed in the BoM-based methods. For instance, only a total of 768 bits are extracted from 2000 (D_r) components when *HQ+MEANPOOL* is applied, which means that there are at least 1232 components with no bit allocated.

Unlinkability: Resistance against Linkage Attack

The unlinkability test follows the procedure described in section 4.6.4. Table 6.4 shows the separability between same-key matching distribution and different-key matching distribution, as well as the statistical information of the two distributions. The experimental results echo with the recognition accuracy of the algorithms in section 6.4.2. While the different-key distributions of the algorithms are almost identical to each other, the same-key distributions become the determining factor of the separability. Same-key genuine distribution with higher mean and lower variance usually results in better separation with the impostor distribution (i.e. better recognition accuracy) and the different-key genuine distribution (i.e. better unlinkability).

Table 6.4: Separability of the proposed cancellable fixed-length representation expressed in the form of "separability$(\mu_{SKG}, \sigma^2_{SKG})[\mu_{DKG}, \sigma^2_{DKG}]$". mu_{SKG} and σ^2_{SKG} represent the mean and variance of the same-key genuine matching distribution, while mu_{DKG} and σ^2_{DKG} are the equivalent parameters of the different-key genuine matching distribution. Since the decimal values shown are rounded to the nearest 0.01, any value that is less than 0.005 are written as <0.005.

Algorithm	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
KPCA+ZT	3.66 (0.78,0.01) [0.50,<0.005]	2.82 (0.78,0.02) [0.50,<0.005]	2.29 (0.71,0.01) [0.50,<0.005]	1.95 (0.69,0.02) [0.50,<0.005]
HQ+MEANPOOL+ZT	2.77 (0.63,<0.005) [0.50,<0.005]	3.19 (0.64,<0.005) [0.50,<0.005]	1.68 (0.58,<0.01) [0.50,<0.005]	1.73 (0.58,<0.005) [0.50,<0.005]
SQ+MEANPOOL+ZT	3.19 (0.66,<0.005) [0.50,<0.005]	3.58 (0.67,<0.005) [0.50,<0.005]	2.30 (0.60,<0.005) [0.50,<0.005]	2.36 (0.61,<0.005) [0.50,<0.005]
KPCA+DQ	4.90 (0.85,0.01) [0.50,<0.005]	4.00 (0.86,0.01) [0.50,<0.005]	2.59 (0.72,0.01) [0.50,<0.005]	2.83 (0.73,0.01) [0.50,<0.005]
HQ+MEANPOOL+DQ	3.62 (0.69,0.01) [0.50,<0.005]	3.16 (0.68,0.01) [0.50,<0.005]	2.04 (0.58,<0.005) [0.50,<0.005]	2.29 (0.59,<0.005) [0.50,<0.005]
SQ+MEANPOOL+DQ	4.38 (0.73,0.01) [0.50,<0.005]	3.29 (0.72,0.01) [0.50,<0.005]	2.30 (0.60,<0.005) [0.50,<0.005]	2.36 (0.61,<0.005) [0.50,<0.005]

Entropy: Resistance against Brute Force Attack

The entropy measures the aggregated bit randomness of the final bit-string. A bit is considered completely random if its probability distribution is uniform, i.e. the probability of it being 0 and 1 are equal. Since binary values are discrete values, the differential entropy in (3.6.7) and (3.6.8) are not applicable. The entropy of a binary vector (bit-string) is derived from the discrete entropy in (3.6.6) as shown below

$$\begin{aligned} H(\hat{V}_b) &= -\sum_{i=1}^{S(\hat{V}_b)} P_{(i)}(\hat{V}_b) \log_2 P_{(i)}(\hat{V}_b) \\ &= -[\Pr(\hat{V}_b = 0) \log_2 \Pr(\hat{V}_b = 0) + \Pr(\hat{V}_b = 1) \log_2 \Pr(\hat{V}_b = 1)], \end{aligned} \tag{6.4.2}$$

where $\hat{V}_b$ may be any bit in the bit-string. The total entropy of the bit-string is simply the sum of the entropies of all bits,

$$H(\hat{\mathbf{V}}_b) = \sum_{i=1}^{D_b} H(\hat{V}_{b(i)}). \tag{6.4.3}$$

Furthermore, there is a possibility that one bit in the bit-string is dependent on another bit. To account for this, the entropy is also evaluated based on the first-order dependency tree approximation [234], defined as

$$\tilde{H}(\hat{\mathbf{V}}_b) = H(\hat{\mathbf{V}}_b) - \max_{\forall i,j \in [1,D_b]} (\sum_{i=1}^{D_b} I(\hat{V}_{b(i)}; \hat{V}_{b(j)})), \text{ for } i \neq j, \tag{6.4.4}$$

where $I(\hat{V}_{b(i)}; \hat{V}_{b(j)})$ is the mutual information between a bit $\hat{V}_{b(i)}$ and its parent $\hat{V}_{b(j)}$ in the dependency tree, defined as

$$I(\hat{V}_{b(i)}; \hat{V}_{b(j)}) = \sum_{\hat{V}_{b(i)}} \sum_{\hat{V}_{b(j)}} P(\hat{V}_{b(i)}, \hat{V}_{b(j)}) \log \left[\frac{P(\hat{V}_{b(i)}, \hat{V}_{b(j)})}{P(\hat{V}_{b(i)}) P(\hat{V}_{b(j)})} \right], \tag{6.4.5}$$

In Zhou et al.'s method [234], the dependency tree was pre-trained using the optimization method proposed by Chow and Liu [235] and applied on the testing set to compute the entropy. However, due to the variations between the training set and the testing set, the entropy obtained for the testing set was generally higher and might not be the best estimation. In this section, we run exhaustive search through all dependency combinations on the testing set directly to obtain the worst case entropy.

Table 6.5 shows the entropies of the bit-string estimated both with and without considering bit dependency. From the two entropy estimations, it is evident that the bits are rather independent of each other as the entropy loss in first-order dependency estimation is trivial.

Moreover, it is as usual that longer bit-length produces higher entropy. Nevertheless, the average entropies per bit position ($H(\hat{\mathbf{V}}_b)/D_b$ or $\tilde{H}(\hat{\mathbf{V}}_b)/D_b$) in the bit-strings are similar. For example, for FVC2002 DB1, although the entropies (without first-order dependency) range from 59.55 bits to 1983.50 bits, the average entropies per bit position are all 0.99. This also means that only 1% of the bits lose their randomness. The requirements for such high entropy are both high intra-class bit stability and uniformly distributed inter-class bit values, that is, $\Pr(\hat{V}_b = 0) \approx \Pr(\hat{V}_b = 1) \approx 0.5$.

Resistance against Hill-Climbing Attack

Apart from the security and privacy issues discussed above, the adversary may iteratively present the estimated template to the system and utilize some leaked information, such as the matching score, to refine the estimation until the matching score

Table 6.5: Entropies (in bits) of the proposed cancellable bit-string estimated based on (6.4.3) and (6.4.4) in the format of '$H(\hat{\mathbf{V}}_b)$ ($\tilde{H}(\hat{\mathbf{V}}_b)$)'

Algorithm	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
KPCA+ZT	59.62 (59.25)	59.68 (59.28)	59.55 (59.27)	59.61 (59.29)
HQ+MEANPOOL+ZT	1984.58 (1951.86)	1981.32 (1947.83)	1979.50 (1950.67)	1984.23 (1955.20)
SQ+MEANPOOL+ZT	1984.46 (1950.21)	1977.86 (1941.19)	1983.50 (1956.54)	1984.45 (1954.79)
KPCA+DQ	61.91 (60.31)	61.80 (60.08)	62.60 (61.29)	62.37 (60.98)
HQ+MEANPOOL+DQ	757.56 (728.32)	756.70 (726.33)	761.96 (740.25)	760.88 (737.00)
SQ+MEANPOOL+DQ	1006.65 (964.26)	1004.21 (959.24)	1015.14 (985.57)	1012.82 (978.72)

converges. Such attack is known as the hill-climbing attack. Through this attack, the estimated template would eventually approximate to the enrolled template and be accepted by the system. A possible countermeasure against hill-climbing attack is to protect the matching score by score quantization [236, 237]. In this case, the actual matching score is hidden and only the quantized score is put on the channel for decision making. Non-uniform score quantization has been proven [237] to effectively reduce the success rate of hill-climbing attack while paying off with a slight decrease in the recognition rate of the system.

6.4.4 Computational Complexity

The training process of DQ involves iterative bit stability assessment for every bit position, so the computational complexity of DQ training is bounded by $\mathcal{O}(D_r N_t N_d)$. On the other hand, zero-thresholding requires no training. The complexity of bit-string generation using zero-thresholding and DQ are $\mathcal{O}(D_r)$ and $\mathcal{O}(D_r N_d)$ respectively.

From the other perspective, the CPU runtime of the proposed schemes are also recorded for the computational complexity analysis, as displayed in Table 6.6. First of all, since zero-thresholding does not required training, the training times of the bit-string generation schemes using zero-thresholding (which includes MLC generation for N_t samples and training for S2V transformation) are the same as those shown in Table 4.5 and 5.6. For the training stage of schemes using DQ, the additional steps not only include training for DQ, but also the vector generation via corresponding S2V transformation and

cancellable transformation for the N_t samples. This put on some additional time to the training stage, especially for *KPCA+DQ* as template generation through KPCA is more time-consuming. Even so, the time required for DQ training is insignificant in the entire bit-string generation scheme as depicted in Figure 6.5. Since the number of training samples (N_t) and the maximum number of bits allowed per vector component (N_d) are constant across different algorithms, D_r becomes the primary factor affecting the complexity of DQ training. From Figure 6.5, it is clear that the KPCA-based method (with $D_r = 60$) requires much less time for DQ training than the BoM-based methods (with $D_r = 2000$).

Table 6.6: CPU runtime of the proposed fingerprint bit-string generation scheme, running on MATLAB environment (Windows 7) with an Intel® Core™ i5-2430M 2.40GHz processor. The number of iterations, $I = 50$ is used.

Stage	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
KPCA+ZT				
Training	3083.73s	3649.99s	3325.85s	2499.56s
Template Generation	5.88s	6.60s	5.54s	4.16s
HQ+MEANPOOL+ZT				
Training	1216.88s	1655.82s	1769.55s	1743.08s
Template Generation	1.36s	1.82s	2.04s	2.05s
SQ+MEANPOOL+ZT				
Training	3336.43s	3856.58s	3993.17s	3901.77s
Template Generation	1.17s	1.60s	1.83s	1.86s
KPCA+DQ				
Training	4819.08s	5625.29s	4521.13s	3256.85s
Template Generation	5.88s	6.60s	5.54s	4.16s
HQ+MEANPOOL+DQ				
Training	1365.64	1822.89s	1930.05s	1891.54s
Template Generation	1.36s	1.82s	2.04s	2.05s
SQ+MEANPOOL+DQ				
Training	3372.72s	3893.15s	4030.92s	3938.24s
Template Generation	1.17s	1.60s	1.83s	1.86s

As for the bit-string generation stage, the time taken for template binarization is negligible, regardless of the binarization method used. Therefore, the CPU runtimes shown in Table 6.6 are identical to those shown in Table 4.5 and 5.6.

6.5 Summary

In this chapter, the previously proposed fixed-length cancellable fingerprint templates have been converted into binary form through the two types of biometric template bi-

(a) *KPCA+DQ*.

(b) *HQ+MEANPOOL+DQ*.

(c) *SQ+MEANPOOL+DQ*.

Figure 6.5: CPU runtime breakdown charts of the training stage of the proposed bit-string generation scheme when DQ is used, tested on FVC2004 DB2.

narization methods, namely static quantization and dynamic quantization (DQ). The former was demonstrated through zero-thresholding, while the latter was realized by following the bit allocation technique proposed by Lim et al. [231]. The main difference between the two methods is that static quantization assigns equal number of bits to all vector components whereas its contrary assigns different number of bits to the vector components depending on their discriminability. For this reason, DQ requires a training stage to obtain the discriminability and the number of bits allocated to each individual vector component.

Deterioration in the recognition accuracy was observed when zero-thresholding is used as the template binarization method. Zero-thresholding is merely a signum function that registers the sign of the vector values, hence information loss in the final bit-string is expected. On the other hand, DQ is able to improve the recognition accuracy of the biometric template. This is mainly due to its ability to extract bits from the vector in such a way that the intra-class similarity and inter-class dissimilarity are maximized. It is noteworthy that the best performance acquired by DQ is better than the benchmarking methods and is comparable to other non-cancellable fingerprint bit-string generation schemes as highlighted in Table 6.3.

With the excellent performance of dynamic quantization, there exists a risk of revealing the real vector prior binarization. The helper data stored for DQ consists of the number of bits assigned to each vector component as well as the quantization intervals. This information allows the adversary to narrow down the range of real values in the vector and reduces the complexity of reversing the bit-string. Nonetheless, DQ provides slightly better unlinkability than zero-thresholding because of its high intra-class similarity. Both zero-thresholding and DQ produce bit-strings with high entropy per bit position.

Since zero-thresholding only involves comparison operator, the computational power is certainly negligible in relative to the entire bit-string generation scheme. As for DQ, the training stage requires the calculation of some statistical figures as explained in section 6.3.2, but still is not as time-consuming as the MLC algorithm and the S2V transformation.

Overall, zero-thresholding, albeit simple, may lead to performance degradation. As its counterpart, DQ is a strong performance enhancer without much trade-off. Although DQ process may be reversed with the helper data compromised, it merely gets the adversary to the real vector, which is still protected by all preceding phases. The only drawback is the additional but trivial computational power.

CHAPTER 7

Case Study: Application on Bio-cryptosystems

7.1 Introduction

It has been revealed in Chapter 2 that one of the main challenges of bio-cryptosystems is the leakage of biometric data through the public helper data. Therefore, if the biometric data is protected with another BTP scheme, it would be safe. For this purpose, this thesis proposes to apply cancellable biometrics in bio-cryptosystems to enhance the security and privacy of the systems. In this chapter, such hybrid BTP scheme is realized by amalgamating the proposed cancellable fingerprint template with fuzzy extractor (FE). The said hybrid BTP scheme is hereafter referred to as cancellable fuzzy extractor (CaFE). Part of the work in this chapter has been published [238].

7.2 Preliminaries

7.2.1 Error-Correcting Codes

An error-correcting code is a technique of handling errors in data transmissions through noisy channels. It encodes a message into a codeword with redundancy (i.e. codeword length is longer than message length) and allows bit-flips detection and correction on the receiver end. A $(n,k,2t+1)$ error-correcting code produces a n-bit codeword from a k-bit message and the minimum distance between any two codes is $2t+1$. Note that n is also known as the block length. Such error-correcting code is capable of correct-

ing up to t erroneous bits. The ratio $\frac{k}{n}$ is also called the code rate and there are $n - k$ redundant (or parity) bits in the codeword.

The two main categories of error-correcting codes are block codes and convolutional codes, among which examples of block codes include Hamming codes, BCH codes, Reed-Solomon codes, Hadamard codes, Golay codes and Reed-Muller codes. As the name suggests, a block code is an error-correcting code that encodes and decodes data in individual blocks as opposed to non-block codes which are continuous and unterminated. The characteristics of some of these error-correcting codes are discussed below:

1. Hamming codes: Hamming codes are generalized and extended from the Hamming (7,4,3)-code [239] and allows codewords longer than 7 bits. Figure 7.1 provides a visual illustration on the encoding algorithm of the Hamming(15,11,3)-code. Hamming codes have an advantage of having high code rate, for instance, the code rate of the Hamming(31,26,3)-code reaches ≈ 0.84. However, the downside is that since the minimum distance is always 3, it can only correct 1 bit of error per block. It is extremely useful in computer memories [240], where errors are rare. In the case of biocrypto-systems, the errors in a biometric bit-string are usually more than one bit. Even if the bit-string can be divided into multiple blocks, errors in two adjacent bit positions can be hardly corrected.

2. BCH codes: BCH codes form a family of cyclic codes, where one codeword is another codeword circularly shifted. The main advantage of BCH codes is that for the same block length, the number of correctable errors (t) is adjustable, hence multiple bits of error correction is possible within a block. The single-bit error-correcting BCH codes are equivalent to the Hamming codes. In the context of bio-cryptosystems, the parameters of the BCH codes can be selected according to the stability of the biometric bit-string to provide better key stability. In addition, BCH codes can be decoded through syndrome decoding which is more efficient than typical minimum distance decoding.

3. Reed-Solomon codes: Reed-Solomon codes are non-binary error-correcting codes. Like BCH codes, Reed-Solomon codes are able to correct multiple errors in the received codeword. The distinctive difference of reed-solomon codes from the two aforementioned error-correcting codes is that they are designed for correcting burst errors, which are errors that occur in numerous consecutive bits.

Bit position		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Parity bit calculation
Encoded data bits		P1	P2	D1	P3	D2	D3	D4	P4	D5	D6	D7	D8	D9	D10	D11	
Parity bit coverage	P1	✓		✓		✓		✓		✓		✓		✓		✓	D1⊕D2⊕D4⊕D5⊕D7⊕D9⊕D11 = 1⊕0⊕1⊕1⊕0⊕0⊕1 = 0
	P2		✓	✓			✓	✓			✓	✓			✓	✓	D1⊕D3⊕D4⊕D6⊕D7⊕D10⊕D11 = 1⊕0⊕1⊕0⊕0⊕0⊕1 = 1
	P3				✓	✓	✓	✓					✓	✓	✓	✓	D2⊕D3⊕D4⊕D8⊕D9⊕D10⊕D11 = 0⊕0⊕1⊕1⊕0⊕0⊕1 = 1
	P4								✓	✓	✓	✓	✓	✓	✓	✓	D5⊕D6⊕D7⊕D8⊕D9⊕D10⊕D11 = 1⊕0⊕0⊕1⊕0⊕0⊕1 = 1
Codeword		0	1	1	1	0	0	1	1	1	0	0	1	0	0	1	

Figure 7.1: Example of a Hamming(15,11,3)-code given the message '10011001001'. The *P*'s represents parity bits and the *D*'s represents data bits. The parity bits are inserted into the bit positions that are powers of two. Viewing the bit positions as binary numbers, the first parity bit (*P1*) is calculated by XOR-ing all data bits with bit positions which have the least significant bit set (i.e. least significant bit is equivalent to 1). The second parity bit (*P2*) uses the bit positions which have the second least significant bit set and so on.

4. Convolutional codes: A convolutional encoder is often explained as a linear time-invariant (LTI) system. The codeword is generated by performing convolution of the message with the encoder's impulse response. Convolutional codes are highly flexible as they have arbitrary block length and code rate. Unlike block codes, convolutional codes may be decoded with soft-decision decoder. Soft-decision decoder makes decision based on Euclidean distance while hard-decision decoder (for block codes decoding) uses Hamming distance. In this way, soft-decision decoder calculates the likelihood of two codewords being alike using multiple bits in the codewords rather than treating each bit independently and making binary decision of whether two corresponding bits from two codewords are equal (hard-decision encoder). Soft-decision decoder often performs better than its hard-decision counterpart, with the drawback of higher computational cost.

7.2.2 Galois Field Notations

A Galois field, also known as a finite field, is a field that contains a finite number of elements. For example, GF(2) is a Galois field of two elements, {0,1} and the arithmetic operations in GF(2) are modulo-2-based, such as $X + X = 0$ or $X \cdot X = X$. Since the secure sketch involves only binary numbers, GF(2^m) is of interest here.

Let α be a primitive element of GF(2^m), the elements in the Galois field include $\{0, 1, \alpha, \alpha^2, ..., \alpha^{2^{m-2}}\}$ and the primitive polynomial of degree m over GF(2^m) can be expressed as

$$\mathbf{p}(X) = p_{(m-1)}X^{m-1} + p_{(m-2)}X^{m-2} + ... + p_{(1)}X + p_{(0)}, \tag{7.2.1}$$

where $p_{(i)} \in \text{GF}(2)$.

Another important notation in Galois field is the minimal polynomials. Minimal polynomials of GF(2^m) are monic polynomials[1] of smallest degree with coefficients in GF(2). A minimal polynomial is mathematically defined as

$$\mathbf{\Lambda}(X) = \Lambda_a X^a + ... + \Lambda_{(1)}X + \Lambda_{(0)}, \tag{7.2.2}$$

for $\Lambda_{(i)} \in \text{GF}(2)$ and a is the smallest integer that satisfies $\mathbf{\Lambda}(\alpha)|_{\alpha \neq 0} = 0$. A minimal polynomial must also be prime, that is, it cannot be factorized into polynomials of

[1] A monic polynomial is a polynomial with the coefficient of the highest power of variable being 1.

lower degree. For example, the primitive element α in GF(2^4) has two minimal polynomials, $X^4 + X^3 + 1$ and $X^4 + X + 1$; whereas $X^4 + X^2 + 1$ is not a minimal polynomial as it can be further factorized into $(X^2 + X + 1)^2$.

7.3 Nomenclature

Symbol	Description
$\hat{\mathbf{V}}_b \in \{0,1\}^{D_b}$	cancellable fingerprint bit-string
D_b	final bit-length — dimension of the cancellable bit-string
κ_c	user-specific template revocation key
$\mathbf{Q}$	user-specific quantization helper data, including the number of bits and quantization intervals for each vector component
$\mathbf{\Omega}_{\text{train}}$	training MLC templates for KPCA
$\mathbf{C}$	dictionary for BoM transformation
λ	sparsity parameter for BoM transformation
$\mathbf{E}$	error-correcting codeword for secure sketch construction
$\boldsymbol{\kappa}_s$	randomly chosen message for secure sketch construction
$\mathbf{SS}$	public sketch
κ_e	random seed for randomness extractor
Symbol	Description
$\mathbf{R}_{FE}$	random cryptographic key generated by the fuzzy extractor
$\hat{\mathbf{V}}'_b$	query fingerprint bit-string
$\mathbf{E}'$	received error-correcting codeword for cryptographic key reproduction
t	correcting capability of error-correcting codes
n	message length of error-correcting codes

k	codeword/block length of error-correcting codes
α	prime element of a GF
$\mathbf{g}(X)$	generator polynomial of error-correcting codes
$\mathbf{\Lambda}(X)$	minimal polynomials of the elements in a GF
$\mathbf{e}(X)$	errors in the received codeword
Syn	syndrome generated for error-correcting decoding
H	parity check matrix for error-correcting decoding

7.4 The CaFE

In order to visualize the concept of CaFE, the block diagram of FE in Figure 2.2 is redrawn in Figure 7.2. First, the biometric feature (minutiae set) and the user-specific revocation key (κ_c) are used to generate the cancellable bit-string ($\hat{\mathbf{V}}_b$). The code-offset construction of secure sketch [2] is adopted in the proposed framework. In the secure sketch block, an error-correcting codeword ($\mathbf{E}$) generated based on a randomly chosen message ($\boldsymbol{\kappa}_s$) is mixed with the cancellable bit-string to form the sketch ($\mathbf{SS}$), i.e. $\mathbf{SS} = \mathbf{E} \oplus \hat{\mathbf{V}}_b$. Ultimately, the cancellable bit-string is used to extract a highly random and uniformly distributed key/password ($\mathbf{R}_{FE}$) via a randomness extractor with random seed (κ_e). The random key/password generated can be used for data encryption or any other security applications. The randomness extractor is essentially universal hashing [2] by using any family of one-way hash functions. In this thesis, we perform the experiments using the secure hash algorithm (SHA). Beisdes the helper data for fingerprint bit-string generation, the sketch and the random seed for extractor are also stored for key reproduction. The biometric revocation key is held by the user himself as suggested in Chapter 3.

During key reproduction, a query biometric feature and the revocation key are to be presented to the system. Then, a cancellable bit-string ($\hat{\mathbf{V}}'_b$) is regenerated based on these inputs. The recovered codeword ($\mathbf{E}'$) is computed by finding the difference between the query biometric bit-string and the sketch is computed, i.e. $\mathbf{E}' = \mathbf{SS} \oplus \hat{\mathbf{V}}'_b$.

[2] To use universal hashing is to randomly select a hash function from a family of hash functions.

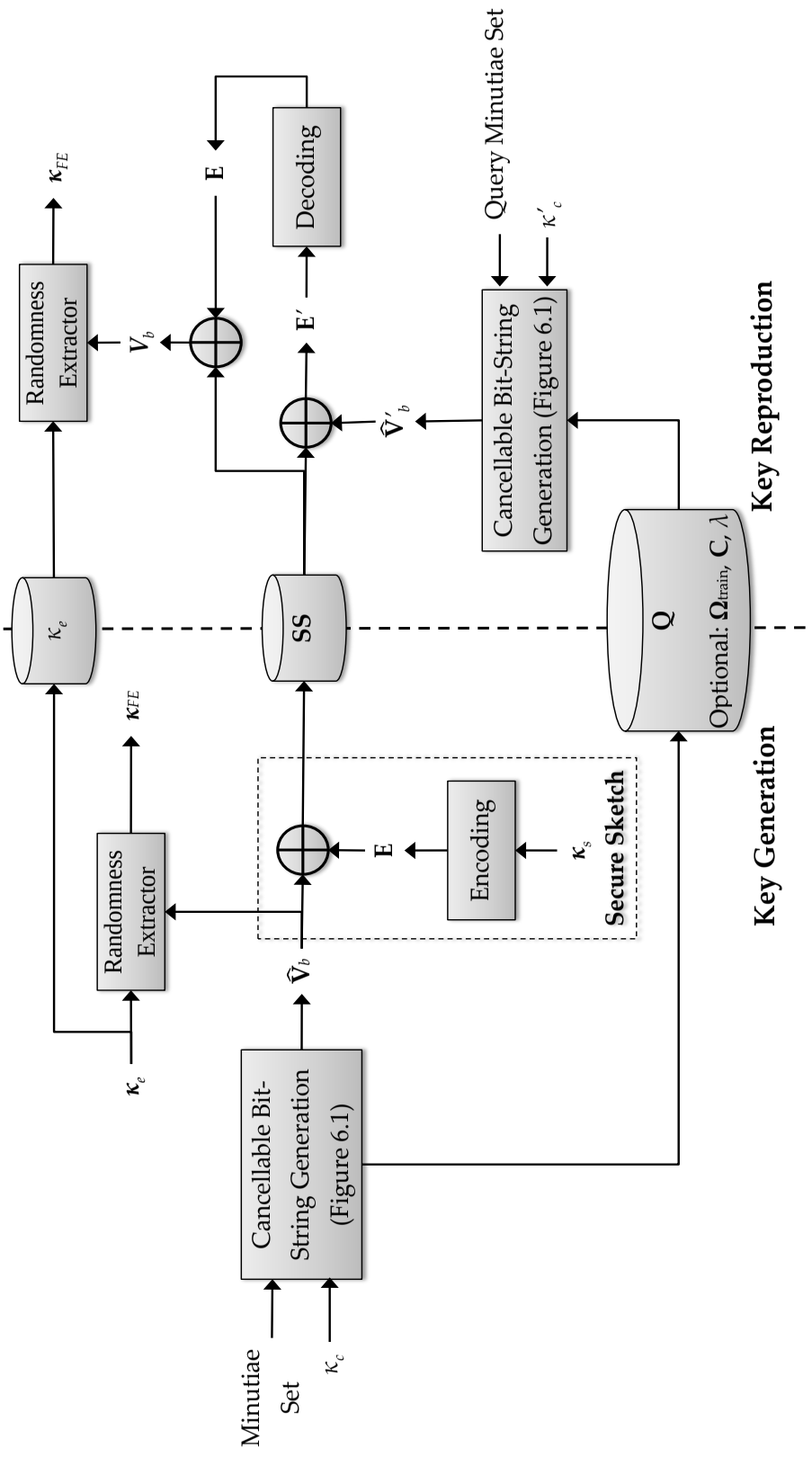


Figure 7.2: The complete framework of the proposed CaFE.

If the recovered codeword is close enough to the true codeword, the true message, and subsequently the original biometric bit-string can be reconstructed through error correction. Finally, the random password is reproduced with the same randomness extractor.

7.4.1 Code-Offset Secure Sketch

The main idea of the code-offset construction of secure sketch is that, given the sketch, if the query biometric bit-string is close enough to the original bit-string in Hamming distance, the true message can be reconstructed. In this case, the maximum allowance of bit-flips in the query bit-string is t bits ($\|\hat{\mathbf{V}}_b - \hat{\mathbf{V}}'_b\|_0 \leq t$) for exact recovery. Out of the error-correcting codes discussed in section 7.2.1, the BCH codes are chosen to construct the secure sketch for the simplicity of encoding/decoding algorithm and flexibility in error-correcting capability.

Encoding BCH Codes

The encoding/decoding process of BCH codes can be explained in the GF(2^m) sense. Given a message $\boldsymbol{\kappa}_s(X)$, one first needs to know the generator matrix of the encoder corresponding to the error-correcting capability, t. It is computed by taking the least common multiple of the minimal polynomials of the first $2t$ elements in GF(2^m) and is mathematically expressed as

$$\mathbf{g}(X) = \text{LCM}(\boldsymbol{\Lambda}_{(1)}(X), \boldsymbol{\Lambda}_{(2)}(X), ..., \boldsymbol{\Lambda}_{(2t)}(X)), \tag{7.4.1}$$

of which $\Lambda_{(i)}(X)$ is the minimal polynomial of the element α^i. The codeword is generated by appending the remainder of dividing $X^{n-k}\boldsymbol{\kappa}_s(X)$ by $\mathbf{g}(X)$ to the message:

$$\mathbf{E}(X) = X^{n-k}\boldsymbol{\kappa}_s(X) + \mathbf{r}(X), \tag{7.4.2}$$

where $X^{n-k}\boldsymbol{\kappa}_s(X) = \mathbf{q}(X)\mathbf{g}(X) + \mathbf{r}(X)$, and $\mathbf{q}(X)$ and $\mathbf{r}(X)$ represents the quotient and remainder of the division operation. An example of BCH encoding is presented in Appendix C.

Decoding BCH Codes

In the proposed CaFE implementation, syndrome decoding is used as the decoding algorithm for BCH codes. Algorithm 7.1 depicts the flow of the syndrome decoding process. Provided that the received codeword is $\mathbf{E}'(X) = \mathbf{E}(X) + \mathbf{e}(X)$, where $\mathbf{e}(X)$ is the errors introduced in the biometric bit-string. The syndrome is defined as

$$\begin{aligned} \mathbf{Syn} &= \mathbf{E}'(X) \cdot \mathbf{H}^{\mathrm{T}} \\ &= (\mathbf{E}(X) + \mathbf{e}(X)) \cdot \mathbf{H}^{\mathrm{T}} \\ &= \mathbf{e}(X) \cdot \mathbf{H}^{\mathrm{T}}, \end{aligned} \tag{7.4.3}$$

where $\mathbf{H}$ is called the parity check matrix

$$\mathbf{H} = \begin{bmatrix} 1 & \alpha & \alpha^2 & \cdots & \alpha^{n-1} \\ 1 & (\alpha^2)^1 & (\alpha^2)^2 & \cdots & (\alpha^2)^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & (\alpha^{2t})^1 & (\alpha^{2t})^2 & \cdots & (\alpha^{2t})^{n-1} \end{bmatrix}, \tag{7.4.4}$$

such that $\mathbf{E}(X) \cdot \mathbf{H}^{\mathrm{T}} = 0$. Therefore the ith syndrome is

$$\mathrm{syn}_{(i)} = \sum_{j=1}^{n-1} e_j \alpha^{i \cdot j}. \tag{7.4.5}$$

If there are non-zero syndromes, then there are errors. The next step is to determine the error locator polynomial, $\sigma(X)$ via the Berlekamp-Massey algorithm [241]. By factorizing the error locator polynomial, the degrees of the roots indicate the locations of the errors from the most significant bit. The roots can be obtained through the Chien search [242]. Once the errors are defined, the true codeword can be acquired by subtracting the errors from the received codeword. If the number of errors detected is inappropriate, say greater than t, then the recovery of the original codeword would fail. Refer to Appendix D for a working example on BCH decoding.

Algorithm 7.1: Syndrome decoder for BCH codes. The superscripts in parentheses indicate the iteration number.

Data: $\mathbf{E}'(X)$
Result: $\mathbf{E}$
begin
  calculate the syndrome based on (7.4.3) and (7.4.5)
  /* The Berlekamp-Massey Algorithm */
  /* Initialization */
  $\boldsymbol{\sigma}^{(0)}(X) \longleftarrow 1$ // error locator polynomial
  $\mathbf{B}^{(0)}(X) \longleftarrow 1$
  $L^{(0)} \longleftarrow 0$
  /* Loop */
  for $i \leftarrow 1$ *to* $2t$ **do**
    $\Delta^{(i)} \longleftarrow \text{syn}_{(i)} + \sum_{j=1}^{L} \sigma_{(j)} \text{syn}_{(i-j)}$
    if $\Delta^{(i)} \neq 0$ ***AND*** $2L^{(i-1)} \leq i-1$ **then**
      $\delta^{(i)} \longleftarrow 1$
    else
      $\delta^{(i)} \longleftarrow 0$
    end
    $L^{(i)} \longleftarrow \delta(i - L^{(i-1)}) + (1-\delta^{(i)})L^{(i-1)}$
    $\mathbf{B}^{(i)}(X) \longleftarrow \delta^{(i)}(\Delta^{(i)})^{-1}\boldsymbol{\sigma}^{(i-1)}(X) + (1-\delta^{(i)})X\mathbf{B}^{(i-1)}(X)$
    $\boldsymbol{\sigma}^{(i)}(X) \longleftarrow \boldsymbol{\sigma}^{(i-1)}(X) + \Delta^{(i)}X\mathbf{B}^{(i-1)}(X)$
  end
  /* Factorize the error locator polynomial */
  /* N_r: the number of roots */
  /* $\beta_{(n)} \in [1, 2^m - 1]$: the degree of the nth root */
  $\boldsymbol{\sigma}^{(2t)}(X) \longleftarrow \prod_{n=1}^{N_r} (\alpha_{(n)}^{\beta_{(n)}} X - 1)$
  /* Evaluate the error values */
  $\mathbf{e}(X) \longleftarrow \sum_{n=1}^{N_r} X^{2^m-1-\beta_{(n)}}$
  /* Correct errors in the received codeword */
  $\mathbf{E}(X) \longleftarrow \mathbf{E}'(X) - \mathbf{e}(X)$
end

7.5 Experiments and Analyses

7.5.1 Testing Protocol

The training and testing samples allocation follows the description in previous chapters, that is, six out of eight samples per fingerprint are used for training and the remaining two for testing. The training process includes minutiae set to feature vector (S2V) transformation training and dynamic quantization (DQ) training, while the testing process involve the complete CaFE system depicted in Figure 7.2. Both processes

are repeated fivefold with randomly selected samples each trial to obtain average results. The decision making for bio-cryptosystems is binary, so the false rejection rate (FRR) and the false acceptance rate (FAR) are used as the performance indicator instead of the EER and no similarity calculation is required.

Since DQ has been proven (in Chapter 6) to outperform conventional zero-thresholding for biometric template binarization in various perspectives, only the DQ-generated bit-strings are considered in this chapter. Yet the bit-length of the fingerprint bit-string varies with different S2V transformation methods (*KPCA+DQ*, *HQ+MEANPOOL+DQ* and *SQ+MEANPOOL+DQ*), the block length (n) of the secure sketch construction should also vary accordingly. The bit-strings are truncated from the least significant bit whenever necessary.

7.5.2 Performance of the CaFE

Table 7.2 shows the performance of the CaFE system illustrated in Figure 7.2. Since the block length of BCH codes is $n = 2^m - 1$, the fingerprint bit-string is truncated to the same length for secure sketch construction. By truncating the least significant bit from the bit-string, BCH codes with $n = 63$ and $n = 1023$ become adaptable to the *KPCA+DQ* ($D_b = 64$ bits) and *SQ+MEANPOOL+DQ* ($D_b = 1024$ bits) algorithms respectively. As for *HQ+MEANPOOL+DQ* ($D_b = 768$ bits), 3 bits are truncated and the bit-string is divided into three blocks of 255 bits ($n = 255$) each to utilize most of the bits. In addition, BCH codes with $n = 63$ are also implemented on the first 63 bits (only 63 bits out of 768 bits or 1023 bits are used) of *HQ+MEANPOOL+DQ* and *SQ+MEANPOOL+DQ* to compare the results.

From Table 7.2, it is apparent that as the correcting capability decreases, the FRR increases while the FAR reacts otherwise. The FRR and FAR are in contrast to each other, when less errors are being corrected, the chance of recovering the true codeword drops, regardless of whether the query is from a genuine user or an impostor. This is equivalent to increasing the similarity threshold for a fingerprint to be accepted in a conventional biometric authentication system. As bio-cryptosystems are biometric applications of high security requirement, it is more practical to adjust the correcting capability so that FAR = 0% to avoid potential key reproduction by impostors.

Furthermore, for *HQ+MEANPOOL+DQ* and *SQ+MEANPOOL+DQ*, the performances are better when only 63 bits of the bit-string are used compared to when bit-string

Table 7.2: Recognition accuracy (in terms of FRR/FAR in %) of the proposed CaFE compared to other existing bio-cryptosystems. In the table, BCH(n,k,$2t+1$) represents the parameters of the BCH codes used.

Algorithm & Parameters	FVC2002 DB1	FVC2002 DB2	FVC2004 DB1	FVC2004 DB2
KPCA+DQ				
BCH(63,7,31) [1]	0/1.11	0/1.08	0/1.09	2/1.15
BCH(63,10,27) [1]	0/0.10	0/0.14	0/0.10	2/0.26
BCH(63,16,23) [1]	**2/0**	**1/0**	**9/0**	**5/0**
BCH(63,18,21) [1]	2/0	2/0	10/0	7/0
BCH(63,24,15) [1]	3/0	3/0	27/0	13/0
HQ+MEANPOOL+DQ				
BCH(255,9,127) [2]	75/0.00	77/0	98/0	98/0
BCH(63,7,31) [1]	12/0.94	7/0.82	53/0.95	47/0.79
BCH(63,10,27) [1]	19/0.11	13/0.11	67/0.07	59/0.11
BCH(63,16,23) [1]	**30/0**	**24/0**	**77/0**	**69/0**
BCH(63,18,21) [1]	37/0	30/0	83/0	76/0
BCH(63,24,15) [1]	60/0	52/0	90/0	86/0
SQ+MEANPOOL+DQ				
BCH(1023,11,511) [1]	25/0.06	25/0.07	90/0.10	82/0.06
BCH(1023,16,295) [1]	27/0	31/0	94/0	85/0
BCH(63,7,31) [1]	5/0.93	5/1.02	43/0.75	32/0.82
BCH(63,10,27) [1]	7/0.13	9/0.14	58/0.13	48/0.08
BCH(63,16,23) [1]	**16/0**	**12/0**	**72/0**	**53/0**
BCH(63,18,21) [1]	23/0	16/0	77/0	57/0
BCH(63,24,15) [1]	42/0	27/0	89/0	80/0
Existing fingerprint hybrid BTP schemes				
Xu and Wang [156]	11/0	13/0	-	-
Other fingerprint key-generation schemes				
Nguyen et al. [243]	-	11/0	-	41/0.22
Yang et al. [244]	8/0.59	6/0.02	-	41/0.22
Nandakumar et al. [143]	-	4/0.004	-	-
Uludag and Jain [245]	-	15.5/0	-	-

[1] only use the first n bits of the biometric bit-string.

[2] only use the first $3n$ bits of the biometric bit-string, divided into three blocks.

truncation is kept minimal. For example, the CaFE, when incorporated with *SQ+MEANPOOL+DQ*, is able to achieve 16% FRR for BCH(63,16,23), but 27% FRR for BCH(1023,16,295) when FAR=0% on FVC2002 DB1. This is because DQ extracts bit-strings such that the bits with higher reliability are selected first. Thus, errors are more likely to occur in the least significant bits. Figure 7.3 shows an example of the distribution of errors over the entire bit-string of 1024 bits. It is obvious that the error occurrence is more frequent in the least significant bits than in the most significant bits.

Since *KPCA+DQ* produces the best EER among the proposed algorithms (see section 6.4.2), it is reasonable that it also gives the best performance when adopted by the CaFE. It is able to achieve FRR of as low as 1% when FAR=0%. Such performance surpasses existing hybrid BTP scheme using fingerprint and is comparable to the state-of-the-art fingerprint key-generation schemes.

7.5.3 Entropy Analysis

The entropy evaluated in previous chapters are called the output entropy. Such entropy denotes the randomness of the system output based on limited number of samples. The randomness extractor in the CaFE generates an output that is independent from the biometric bit-string and is almost uniformly distributed. Therefore, the output entropy is expected to be close to the output length. What is more important for a cryptosystem is the system entropy — the worst case entropy of the system output that takes into account the data dependencies within the system.

The frequently used notation in bio-cryptosystems is the worst-case entropy, or otherwise known as the min-entropy. It is defined as

$$H_\infty(X) = -\log(\max_x \Pr(X = x)). \tag{7.5.1}$$

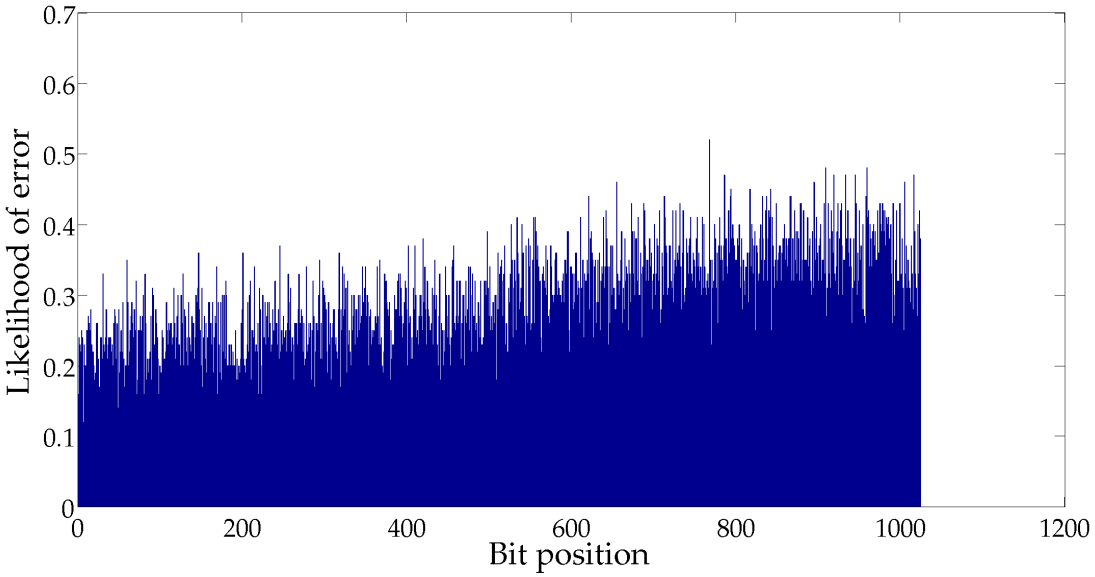


Figure 7.3: Distribution of errors over the bit-string for *SQ+MEANPOOL+DQ* on FVC2002 DB1.

Besides, the average min-entropy of X given Y is given by

$$\tilde{H}_\infty(X|Y) = -\log(\mathbb{E}_{y\leftarrow Y}[2^{-H_\infty(X|Y=y)}]). \tag{7.5.2}$$

By the definitions above, for a fuzzy extractor built from a code-offset secure sketch, Dodis et al. [2] has deduced that the entropy of the key generated ($\boldsymbol{\kappa}_{FE}$) given the public sketch ($\mathbf{SS}$) is

$$\tilde{H}_\infty(\boldsymbol{\kappa}_{FE}|\mathbf{SS}) \geq H_\infty(\hat{\hat{\mathbf{V}}}_b) + k - n - 2\log(\frac{1}{\epsilon}). \tag{7.5.3}$$

In randomness extractor the output ($\boldsymbol{\kappa}_{FE}$) is truly random if the input ($\hat{\hat{\mathbf{V}}}_b$) is random. However, this assumption may not apply to most biometric bit-strings. So, $\boldsymbol{\kappa}_{FE}$ is described as ϵ-close to uniform distribution and the entropy loss $2\log(\frac{1}{\epsilon})$ is included in the formula.

If a variable is taken from a uniform distribution, it yields an entropy of 1. Therefore, the entropy is one way of measuring the "closeness" of one probability distribution to uniform distribution [246]. Experiment shows that the average entropy per bit position (as described in (6.4.2)) of the proposed bit-string for all datasets and all algorithms was found to be $0.99 < H(\hat{V}_b) < 1$ (or $0.99 < \epsilon < 1$). For this, $2\log(\frac{1}{\epsilon})$ is less than 0.02 and is negligible. Also, it has been shown in section 6.4.3 that the fingerprint bit-strings can retain 0.99 of the entropy per bit position. Thus, the min-entropy of the proposed CaFE is $\tilde{H}_\infty(\boldsymbol{\kappa}_{FE}|\mathbf{SS}) \geq 0.99n + k - n = k - 0.01n$ bits. Table 7.3 shows the min-entropies with different error-correcting parameters. By relating Table 7.2 and 7.3, one can conclude that there exists a trade-off between performance and entropy of a bio-cryptosystem. From the expression of $\tilde{H}_\infty(\boldsymbol{\kappa}_{FE}|\mathbf{SS})$ derived above, it is plain that the message length (k) needs to be increased in order to acquire higher entropy. However, longer message also means lower error-correcting capability (t) and directly affects the FRR of the system.

Table 7.3: Min-entropy of the CaFE for different error-correcting parameters.

Parameters	$\tilde{H}_\infty(\boldsymbol{\kappa}_{FE}\|\mathbf{SS})$
BCH(63,7,31)	6
BCH(63,10,27)	9
BCH(63,16,23)	15
BCH(63,18,21)	17
BCH(63,24,15)	23
BCH(255,9,127)	19
BCH(1023,11,511)	1
BCH(1023,16,295)	6

7.6 Summary

In this chapter, an integration between the two models of biometric template protection (cancellable biometrics and bio-cryptosystems) for fingerprint biometrics has been realized, namely the CaFE. Brief descriptions of the two major sections of the proposed scheme are as below:

1. Cancellable biometrics: the goal of this section of CaFE is to generate a cancellable bit-string from fingerprint. In the proposed scheme, *i*) the minutiae extraction is done by using VeriFinger SDK [172]; *ii*) after that, the minutiae are transformed into a set of minutia vectors using the MLC algorithm (3); *iii*) the variable-size and unordered minutia vectors (MLC) are then converted into fixed-length and ordered feature vector with either the KPCA-based method or the BoM-based method (includes hard quantization and soft quantization); *iv*) the cancellable transformation through RP is applied onto the feature vector; *v*) finally, bit-string is produced via DQ. These are the five phases of the cancellable fingerprint generation section.

2. Bio-cryptosystem: the bio-cryptosystem section is demonstrated through the fuzzy extractor [2]. Within it, are the secure sketch that helps to recover the fingerprint bit-string during key reproduction and the randomness extractor that generates the cryptographic key. The code-offset construction of secure sketch with BCH coding was adopted in the experiments.

Experimental results showed that the KPCA-based S2V transformation produces the best performance among the three methods tested. This is expected as the said method also yields the most promising recognition accuracy as a cancellable biometrics. Looking at FVC2002 DB2, the *KPCA+DQ*-based CaFE is able to achieve 1% FRR at FAR = 0%. This result overtakes the benchmarking key-generation bio-cryptosystems. However, trade-off between performance and security (entropy) was observed. The entropy is merely 15 bits corresponding to the aforementioned FRR. To take it further, the proposed CaFE can set at 23 bits of worst-case entropy with 3% FRR for FVC2002 DB2. That is equivalent to a brute-force attack complexity of 2^{23}. This trade-off becomes one of the hindrances of bringing the CaFE into real-life applications [247].

CHAPTER 8

Conclusions and Future Work

8.1 Summary of Thesis Chapters

In this thesis, a unique cancellable fingerprint template generation scheme has been presented. The proposed scheme does not only function as a BTP method standalone, but the output is also adaptable to bio-cryptosystems. A complete framework of generating the cancellable template from fingerprint minutiae was elaborated and evaluated phase-by-phase throughout the thesis. As a finishing touch, the application of the template in bio-cryptosystems was also demonstrated.

A thorough study about the existing fingerprint matching algorithms has been done (Chapter 2). The conclusion was that minutiae-based matching is more robust than texture-based matching due to its ability to address both local feature stability and global feature uniformity. Fixed-radius minutia descriptors was found to stand out among the minutiae-based methods as they are more effective in handling non-linear distortions, and missing and spurious minutiae. Besides, the second part of the literature review covered the works on BTP schemes. For cancellable biometrics particularly, non-invertible transforms using invariant features, although eliminate the need for fingerprint pre-alignment, are subjected to loss of information. On the other hand, biometric salting excels in information-preserving, but is of high risk against reverse attack when the user-specific cancellable key is compromised.

Based on the findings above, a cancellable fingerprint template generation scheme incorporating fixed-radius minutia descriptor that is non-invertible and biometric salting which provides revocability was proposed (Chapter 3). The proposed descriptor, namely the MLC, generates a minutia vector by observing the distribution of the neigh-

bouring minutiae along multiple lines centred at the reference minutia. The MLC is designed to be invariant to rotations and translations. In addition, the adverse effects caused by scaling and non-linear local distortions can be minimized by adjusting the radius size. Two distinct branches of the MLC algorithm were proposed, including the one that counts the number of neighbouring minutiae within the radius (*MLCN*) and the one that calculates the mean distance of the neighbouring minutiae from the centre (*MLCD*). The objective of cancellable biometrics is not accomplished without revocability. Two cancellable transformations, namely RP and permutation, are borrowed from the concept of biometric salting to complete the cancellable template generation scheme.

Experimental results showed that *MLCD* performs better than *MLCN*. Besides, RP suffers from performance deterioration while permutation is able to maintain the recognition accuracy. Overall, the proposed scheme produced EER comparable to the existing methods. Furthermore, the security and privacy of the MLC algorithm was analysed. Permutation has zero privacy strength once the cancellable template and the user-specific revocation key are compromised. Although RP introduces dimensionality reduction, it also provides very weak privacy in the context since MLC is sparse. In spite of that, the raw minutiae are still protected by the MLC algorithm. It was shown that the MLC algorithm is mathematically irreversible even though it may reduce the complexity of brute-force attack. The proposed method also yields strong unlinkability and is computationally inexpensive. Note that the experiments are performed in the MATLAB environment on Windows 7 with an Intel® Core™ i5-2430M 2.40GHz processor. The CPU runtimes shown are expected to shorten in real-life application with more efficient platform.

As the number and positions of the minutiae in a fingerprint may vary each time it is scanned, the cancellable template produced by the MLC algorithm is variable in size and unordered. This hinders the application of the fingerprint template for various practical purposes such as bio-cryptosystems, sophisticated classifiers like SVM, dynamic quantization for template binarization, standard vectors comparison metrics and continuous classification for fingerprint indexing. Therefore, we introduced the S2V transformation via kernel subspace analysis (Chapter 4) to convert the said template into a fixed-length and ordered vector. In this context, PCA was chosen as the most suitable subspace analysis model, onto which the kernel method is applied to form KPCA. A kernel function was specially designed based on the matching func-

tion of MLC templates so that it can be directly applicable to the previously proposed fingerprint template.

The KPCA-based S2V transformation was shown to yield remarkable performance. However, the training samples stored leak information about the user's original template. Yet, the raw minutiae remain protected as the training samples are generated by the MLC algorithm which is designed to be mathematically non-invertible. There are multiple layers of protection to break before unveiling the minutiae information. Besides, the vectorized template showed high value in other aspects of security and privacy such as unlinkability and entropy. Although the S2V transformation may put on additional computational power to the framework, most operations are executed off-line. The on-line runtime is within acceptable range considering the current computer technology.

Further, a second approach towards S2V transformation based on the BoM paradigm was presented (Chapter 5). The concept of BoM modelling, originated from the BoW paradigm, was consolidated with three design perspectives, including minutiae representation, space partitioning and minutiae labelling. In the experiments, two design choices of minutiae labelling, viz. hard quantization and soft quantization, are implemented with K-means clustering and dictionary learning respectively. Results showed that the latter exhibits lower EER than the former due to its fuzziness in labelling/assigning the minutiae. By allowing a minutia to be assigned to more than one atom, the soft quantization algorithm is able to reduce the quantization error. In terms of security and privacy, soft quantization takes more effort to crack, and the produced cancellable template has slightly higher unlinkability than its counterpart. The drawback is that soft quantization requires more than doubled the training time of hard quantization.

In most of the aforementioned practical applications of fixed-length and ordered fingerprint representation, binary input is also a vital requirement. Therefore, the proposed cancellable template was binarized into bit-string (Chapter 6). Note that bit-string refers to fixed-length binary vector. Both static quantization (or specifically zero-thresholding) and DQ were used to realize fingerprint template binarization in this thesis for comparison. Apparently, DQ outperforms zero-thresholding by a significant extent. This is because DQ aims at maximizing the inter-class similarity and minimizing intra-class similarity while allocating the bits. However, the helper data stored for DQ may leak useful information about the values of the real vector. Both binarization

techniques have similar unlinkability and entropy. From the aspect of computational complexity, DQ requires additional computational power for training, but both techniques are equally time-efficient during template generation.

Now that a complete cancellable fingerprint bit-string generation scheme has been developed, it can be a "plug and play" component to any appropriate application besides a typical fingerprint authentication system. In this thesis, the application of the proposed fingerprint bit-string in a key-generation bio-cryptosystem was demonstrated (Chapter 7). To be more precise, the cancellable fingerprint was amalgamated with fuzzy extractor to form a hybrid BTP, known as the CaFE. In addition to the bit-string generation scheme, the CaFE consists of two other components, i.e. secure sketch with code-offset construction and randomness extractor. Experimental results showed that the CaFE is able to achieve FRR as low as 2% when FAR = 0% which is comparable to the state-of-the-art key-generation bio-cryptosystems. The output entropy could preserve more than 99% of the output length, but the worst-case system entropy appeared to be a lot weaker. Nevertheless, the system entropy can be strengthened by sacrificing the performance while maintaining it at acceptable range.

8.2 Concluding Remarks

Although two cancellable transformations were used, RP was proven to be more superior than permutation. Also, DQ was preferable to zero-thresholding as the template binarization technique for its excellent performance while no significant additional on-line runtime required. The entire cancellable bit-string generation scheme is narrowed down to the choice of S2V transformation with three options available, including the KPCA-based method, hard quantization-based BoM modelling and soft quantization-based BoM modelling. Figure 8.1 depicts a triadic analysis on the three important aspects of the methods. The trade-offs among the three aspects can be easily observed from the figure. The KPCA-based method provides extremely high performance, but lacks security strength and computational efficiency. On the other hand, soft quantization-based BoM modelling compromises on the recognition accuracy to offer lower on-line computational power demand and better resistance against various attacks. Hard quantization-based BoM modelling however, shows poorer results than soft quantization in all aspects. In conclusion, the end product of this thesis can accommodate to various applications, ranging from general biometric authentication systems

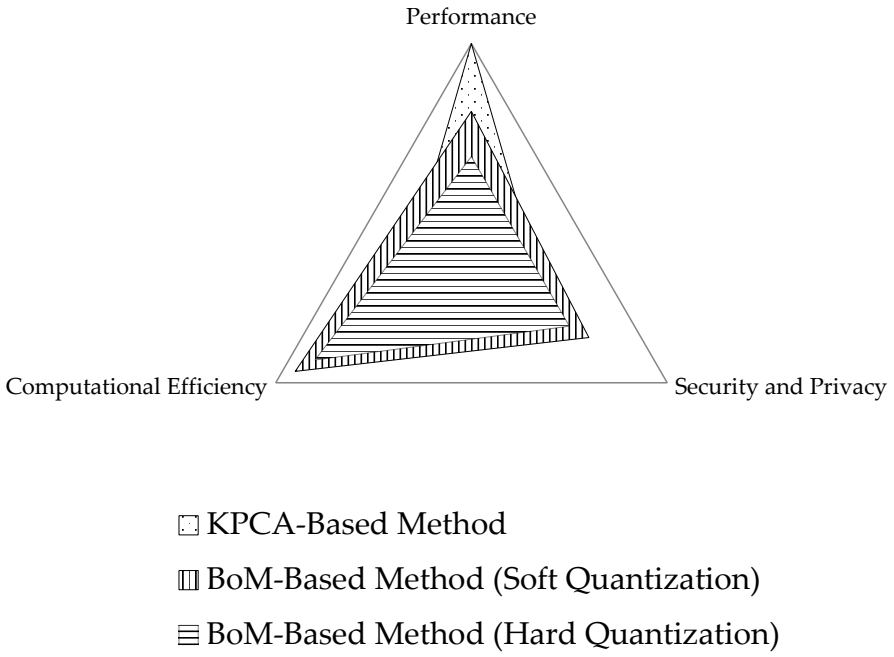


Figure 8.1: Radar chart on the performance, security and privacy, and computational efficiency of the proposed S2V transformation methods. The magnitude of each of the aspects represents its strength.

to highly secured cryptosystems, with proper choice of method and parameters. What is more interesting is that the two major contributions of this study, including the MLC descriptor and the two S2V transformation approaches, are mutually exclusive and can each be adopted by any similar biometric system individually.

8.3 Directions for Future Works

In this section, some possible future works of this research project are briefly discussed. They include improvements of algorithms and potential works extended from this study, with the intention to incite the practicality of the proposed framework as well as to discover research opportunities in relevant areas.

In the real world, a biometric system usually allows regular enrolments of new users after the system has already been set up. For training-based algorithms, such as KPCA, *a posteriori* BoM modelling and DQ, the enrolment of a new user may affect the pre-trained helper data. For example, the minutia vectors of a new fingerprint may change the positions of the clusters obtained by K-means algorithm, and subsequently affect the feature vectors of existing users. Therefore, studies on proper update protocols

of the helper data can be a future research direction. Although on-line algorithms for BoM training, such as on-line *K*-means clustering [248, 249], streaming *K*-means clustering [250] and on-line dictionary learning [251,252], have been proposed to incrementally update the clusters/atoms, the problem of updating the existing feature vectors in the database remains unsolved. One convenient and simple solution is to use a large amount of training samples so that the dictionary obtained is globally representative and will not be affected by new enrolment. This is related to the point-proportion admissibility[1] of clustering algorithms.

The issue of limited direct applications faced by minutiae-based fingerprint template has been dealt with in this thesis by introducing two distinct S2V transformation methods as the countermeasures. This study intends to spur researches regarding this less explored yet prominent issue in the realm of fingerprint biometrics. New methods and enhancement of the existing methods towards S2V transformation are foreseeable.

Moreover, the proposed S2V transformation methods are applicable to most descriptor-based fingerprint features, for example MCC [38] and other recently proposed descriptors such as multi-line neighbouring relation-based descriptor [254] and minutiae relation code (MRC) [255]. Therefore, further study on more efficient and robust minutiae representations also helps to improve the proposed cancellable fingerprint generation scheme.

[1] A clustering algorithm is said to be point-proportion admissible if after adding a point that is identical to the existing points in the clusters, the cluster boundaries remain unchanged. [253]

References

[1] L. Wang and M. Dai, "Application of a new type of singular points in fingerprint classification," *Pattern Recognit. Lett.*, vol. 28, no. 13, pp. 1640–1650, 2007.

[2] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Advances in cryptology-Eurocrypt 2004*. Springer, 2004, pp. 523–540.

[3] A. K. Jain, S. C. Dass, and K. Nandakumar, "Soft biometric traits for personal recognition systems," in *Biometric Authentication*. Springer, 2004, pp. 731–738.

[4] J. Li, Y. Wang, T. Tan, and A. K. Jain, "Live face detection based on the analysis of fourier spectra," in *Defense and Security*. International Society for Optics and Photonics, 2004, pp. 296–303.

[5] E. C. Lee, K. R. Park, and J. Kim, "Fake iris detection by using purkinje image," in *Advances in Biometrics*. Springer, 2005, pp. 397–403.

[6] D. Baldisserra, A. Franco, D. Maio, and D. Maltoni, "Fake fingerprint detection by odor analysis," in *Advances in Biometrics*. Springer, 2005, pp. 265–272.

[7] S. Biswas, K. W. Bowyer, and P. J. Flynn, "A study of face recognition of identical twins by humans," in *Information Forensics and Security (WIFS), 2011 IEEE International Workshop on*. IEEE, 2011, pp. 1–6.

[8] P. J. Phillips, P. J. Flynn, K. W. Bowyer, R. W. V. Bruegge, P. J. Grother, G. W. Quinn, and M. Pruitt, "Distinguishing identical twins by face recognition," in *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*. IEEE, 2011, pp. 185–192.

[9] "MS Windows NT kernel description," http://www.jpn.gov.my/en/informasi/pengenalan-kepada-mykad, accessed: July 29, 2015.

[10] E. R. Henry, *Classification and uses of Fingerprints*, Routledge, London, 1900.

[11] A. K. Jain, Y. Chen, and M. Demirkus, "Pores and ridges: high-resolution fingerprint matching using level 3 features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 15–27, 2007.

[12] M. Kass and A. Witkin, "Analyzing oriented patterns," *Comput. Vis. Graph. Image Process.*, vol. 37, no. 3, pp. 362–385, 1987.

[13] N. K. Ratha, S. Chen, and A. K. Jain, "Adaptive flow orientation-based feature extraction in fingerprint images," *Pattern Recognit.*, vol. 28, no. 11, pp. 1657–1672, 1995.

[14] A. Jain, L. Hong, and R. Bolle, "On-line fingerprint verification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 4, pp. 302–314, 1997.

[15] A. M. Bazen and S. H. Gerez, "Systematic methods for the computation of the directional fields and singular points of fingerprints," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 905–919, 2002.

[16] Y. Wang, J. Hu, and H. Schroder, "A gradient based weighted averaging method for estimation of fingerprint orientation fields," in *Digital Image Computing: Techniques and Applications, 2005. DICTA'05. Proceedings 2005*. IEEE, 2005, pp. 29–29.

[17] J. Zhou and J. Gu, "A model-based method for the computation of fingerprints' orientation field," *IEEE Trans. Image Process.*, vol. 13, no. 6, pp. 821–835, 2004.

[18] S. Ram, H. Bischof, and J. Birchbauer, "Modelling fingerprint ridge orientation using legendre polynomials," *Pattern Recognit.*, vol. 43, no. 1, pp. 342–357, 2010.

[19] L. O'Gorman and J. V. Nickerson, "An approach to fingerprint filter design," *Pattern Recognit.*, vol. 22, no. 1, pp. 29–38, 1989.

[20] L. Ji and Z. Yi, "Fingerprint orientation field estimation using ridge projection," *Pattern Recognit.*, vol. 41, no. 5, pp. 1491–1503, 2008.

[21] L. Hong, Y. Wan, and A. Jain, "Fingerprint image enhancement: algorithm and performance evaluation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 777–789, 1998.

[22] S. Greenberg, M. Aladjem, D. Kogan, and I. Dimitrov, "Fingerprint image enhancement using filtering techniques," in *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, vol. 3. IEEE, 2000, pp. 322–325.

[23] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based fingerprint matching," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 846–859, 2000.

[24] W. Zhang, Y. Y. Tang, and X. You, "Fingerprint enhancement using wavelet transform combined with gabor filter," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 18, no. 08, pp. 1391–1406, 2004.

[25] J. Yang, L. Liu, T. Jiang, and Y. Fan, "A modified gabor filter design method for fingerprint image enhancement," *Pattern Recognit. Lett.*, vol. 24, no. 12, pp. 1805–1817, 2003.

[26] W. Wang, J. Li, F. Huang, and H. Feng, "Design and implementation of log-gabor filter in fingerprint image enhancement," *Pattern Recognit. Lett.*, vol. 29, no. 3, pp. 301–308, 2008.

[27] A. Ross, A. Jain, and J. Reisman, "A hybrid fingerprint matcher," *Pattern Recognit.*, vol. 36, no. 7, pp. 1661–1673, 2003.

[28] S. Chikkerur, C. Wu, and V. Govindaraju, "A systematic approach for feature extraction in fingerprint images," in *Biometric Authentication*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2004, vol. 3072, pp. 344–350.

[29] D. Maio and D. Maltoni, "Direct gray-scale minutiae detection in fingerprints," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 1, pp. 27–40, 1997.

[30] X. Jiang, W.-Y. Yau, and W. Ser, "Detecting the fingerprint minutiae by adaptive tracing the gray-level ridge," *Pattern Recognit.*, vol. 34, no. 5, pp. 999–1013, 2001.

[31] A. Farina, Z. M. Kovacs-Vajna, and A. Leone, "Fingerprint minutiae extraction from skeletonized binary images," *Pattern Recognit.*, vol. 32, no. 5, pp. 877–889, 1999.

[32] F. Zhao and X. Tang, "Preprocessing and postprocessing for skeleton-based fingerprint minutiae extraction," *Pattern Recognit.*, vol. 40, no. 4, pp. 1270–1281, 2007.

[33] R. Cappelli, D. Maio, A. Lumini, and D. Maltoni, "Fingerprint image reconstruction from standard templates," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 9, pp. 1489–1503, 2007.

[34] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingeprint Recognition*. London: Springer-Verlag, 2009.

[35] R. M. Bolle, J. H. Connell, and N. K. Ratha, "Biometric perils and patches," *Pattern Recognit.*, vol. 35, no. 12, pp. 2727–2738, 2002.

[36] A. Juels and M. Wattenberg, "A fuzzy commitment scheme," in *Proceedings of the 6th ACM Conference on Computer and Communications Security*. ACM, 1999, pp. 28–36.

[37] B. Dorizzi, R. Cappelli, M. Ferrara, D. Maio, D. Maltoni, N. Houmani, S. Garcia-Salicetti, and A. Mayoue, "Fingerprint and on-line signature verification competitions at icb 2009," in *Proceedings of the International Conference on Biometrics (ICB)*, Alghero, Italy, June 2009, pp. 725–732.

[38] R. Cappelli, M. Ferrara, and D. Maltoni, "Minutia cylinder-code: A new representation and matching technique for fingerprint recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, pp. 2128–2141, 2010.

[39] X. Jiang and W.-Y. Yau, "Fingerprint minutiae matching based on the local and global structures," in *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, vol. 2. IEEE, 2000, pp. 1038–1041.

[40] D. Lee, K. Choi, and J. Kim, "A robust fingerprint matching algorithm using local alignment," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, vol. 3. IEEE, 2002, pp. 803–806.

[41] T.-Y. Jea and V. Govindaraju, "A minutia-based partial fingerprint recognition system," *Pattern Recognit.*, vol. 38, no. 10, pp. 1672–1684, 2005.

[42] S. Chikkerur, A. N. Cartwright, and V. Govindaraju, "K-plet and coupled BFS: a graph based fingerprint representation and matching algorithm," in *Advances in Biometrics*. Springer, 2005, pp. 309–315.

[43] X. Tong, S. Liu, J. Huang, and X. Tang, "Local relative location error descriptor-based fingerprint minutiae matching," *Pattern Recognit. Lett.*, vol. 29, no. 3, pp. 286–294, 2008.

[44] A. K. Hrechak and J. A. McHugh, "Automated fingerprint recognition using structural matching," *Pattern Recognit.*, vol. 23, no. 8, pp. 893–904, 1990.

[45] A. Wahab, S. Chin, and E. Tan, "Novel approach to automated fingerprint recognition," in *Vision, Image and Signal Processing, IEE Proceedings-*, vol. 145, no. 3. IET, 1998, pp. 160–166.

[46] R. C. Gonzalez and R. E. Woods, "Digital image processing," 2002.

[47] J. Bringer and V. Despiegel, "Binary feature vector fingerprint representation from minutiae vicinities," in *Biometrics: Theory Applications and Systems (BTAS), 2010 Fourth IEEE International conference on*, Washington, DC, September 2010, pp. 1–6.

[48] H. W. Kuhn, "The hungarian method for the assignment problem," *Nav. Res. Logist. Q.*, vol. 2, no. 1-2, pp. 83–97, 1955.

[49] M. Tico and P. Kuosmanen, "Fingerprint matching using an orientation-based minutia descriptor," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, pp. 980–992, 2003.

[50] J. Qi and Y. Wang, "A robust fingerprint matching method," *Pattern Recognit.*, vol. 38, no. 10, pp. 1665–1671, 2005.

[51] X. Wang, J. Li, and Y. Niu, "Fingerprint matching using OrientationCodes and PolyLines," *Pattern Recognit.*, vol. 40, no. 11, pp. 3164–3177, 2007.

[52] H. Choi, K. Choi, and J. Kim, "Fingerprint matching incorporating ridge features with minutiae," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 2, pp. 338–345, 2011.

[53] J. Feng, "Combining minutiae descriptors for fingerprint matching," *Pattern Recognit.*, vol. 41, no. 1, pp. 342–352, 2008.

[54] R. Zhou, S. Sin, D. Li, T. Isshiki, and H. Kunieda, "Adaptive SIFT-based algorithm for specific fingerprint verification," in *Hand-Based Biometrics (ICHB), 2011 International Conference on*. IEEE, 2011, pp. 1–6.

[55] R. Zhou, D. Zhong, and J. Han, "Fingerprint identification using SIFT-based minutia descriptors and improved all descriptor-pair matching," *Sens.*, vol. 13, no. 3, pp. 3142–3156, 2013.

[56] G. Aggarwal, R. M. Bolle, T.-Y. Jea, and N. K. Ratha, "Fingerprint representation using gradient histograms," Jan. 7 2014, US Patent 8,625,861.

[57] G. Bebis, T. Deaconu, and M. Georgiopoulos, "Fingerprint identification using Delaunay triangulation," in *Information Intelligence and Systems, 1999. Proceedings. 1999 International Conference on*. IEEE, 1999, pp. 452–459.

[58] G. Parziale and A. Niel, "A fingerprint matching using minutiae triangulation," in *Biometric Authentication*. Springer, 2004, pp. 241–248.

[59] N. Liu, Y. Yin, and H. Zhang, "A fingerprint matching algorithm based on Delaunay triangulation net," in *Computer and Information Technology, 2005. CIT 2005. The Fifth International Conference on*. IEEE, 2005, pp. 591–595.

[60] W. Yang, J. Hu, and M. Stojmenovic, "NDTC: A novel topology-based fingerprint matching algorithm using N-layer Delaunay triangulation net check," in *Industrial Electronics and Applications (ICIEA), 2012 7th IEEE Conference on*. IEEE, 2012, pp. 866–870.

[61] W. Xu, X. Chen, and J. Feng, "A robust fingerprint matching approach: Growing and fusing of local structures," in *Advances in Biometrics*. Springer, 2007, pp. 134–143.

[62] H. Khazaei and A. Mohades, "Fingerprint matching algorithm based on Voronoi diagram," in *Computational Sciences and Its Applications, 2008. ICCSA'08. International Conference on*. IEEE, 2008, pp. 433–440.

[63] R. Soleymani and M. C. Amirani, "A hybrid fingerprint matching algorithm using Delaunay triangulation and Voronoi diagram," in *Electrical Engineering (ICEE), 2012 20th Iranian Conference on*. IEEE, 2012, pp. 752–757.

[64] S. Chikkerur and N. Ratha, "Impact of singular point detection on fingerprint matching performance," in *Automatic Identification Advanced Technologies, 2005. Fourth IEEE Workshop on*. IEEE, 2005, pp. 207–212.

[65] Y. Jie, Z. Renjie, S. Qifa *et al.*, "Fingerprint minutiae matching algorithm for real time system," *Pattern Recognit.*, vol. 39, no. 1, pp. 143–146, 2006.

[66] L. Liu, T. Jiang, J. Yang, and C. Zhu, "Fingerprint registration by maximization of mutual information," *IEEE Trans. Image Process.*, vol. 15, no. 5, pp. 1100–1110, 2006.

[67] J. Gu, J. Zhou, and C. Yang, "Fingerprint recognition by combining global structure and local cues," *IEEE Trans. Image Process.*, vol. 15, no. 7, pp. 1952–1964, 2006.

[68] J. Feng and A. Cai, "Fingerprint representation and matching in ridge coordinate system," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 4. IEEE, 2006, pp. 485–488.

[69] N. Yager and A. Amin, "Fingerprint alignment using a two stage optimization," *Pattern Recognit. Lett.*, vol. 27, no. 5, pp. 317–324, 2006.

[70] P. Das, K. Karthik, and B. C. Garai, "An efficient hybrid fingerprint matching algorithm," in *Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), 2011 Third National Conference on*. IEEE, 2011, pp. 150–153.

[71] A. M. Bazen and S. H. Gerez, "An intrinsic coordinate system for fingerprint matching," in *Audio-and Video-Based Biometric Person Authentication*. Springer, 2001, pp. 198–204.

[72] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "FingerCode: a filterbank for fingerprint representation and matching," in *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, vol. 2. IEEE, 1999.

[73] L. Sha, F. Zhao, and X. Tang, "Improved fingercode for filterbank-based fingerprint matching," in *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, vol. 2. IEEE, 2003, pp. II–895.

[74] ——, "Fingerprint matching using minutiae and interpolation-based square tessellation fingercode," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, vol. 2. IEEE, 2005, pp. II–41.

[75] F. Benhammadi, M. Amirouche, H. Hentous, K. Bey Beghdad, and M. Aissani, "Fingerprint matching from minutiae texture maps," *Pattern Recognit.*, vol. 40, no. 1, pp. 189–197, 2007.

[76] A. B. J. Teoh, D. N. C. Ling, and O. T. Song, "An efficient fingerprint verification system using integrated wavelet and fourier–mellin invariant transform," *Image and Vision Computing*, vol. 22, no. 6, pp. 503–513, 2004.

[77] L. Nanni and A. Lumini, "A hybrid wavelet-based fingerprint matcher," *Pattern Recognit.*, vol. 40, no. 11, pp. 3146–3151, 2007.

[78] A. Chebira, L. P. Coelho, A. Sandryhaila, S. Lin, W. G. Jenkinson, J. MacSleyne, C. Hoffman, P. Cuadra, C. Jackson, M. Puschel *et al.*, "An adaptive multiresolution approach to fingerprint recognition," in *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, vol. 1. IEEE, 2007, pp. I–457.

[79] T. Amornraksa and S. Tachaphetpiboon, "Fingerprint recognition using dct features," *Electron. Lett.*, vol. 42, no. 9, pp. 522–523, 2006.

[80] L. Nanni and A. Lumini, "Local binary patterns for a hybrid fingerprint matcher," *Pattern Recognit.*, vol. 41, no. 11, pp. 3461–3466, 2008.

[81] M.-K. Hu, "Visual pattern recognition by moment invariants," *IRE Trans. Inf. Theory*, vol. 8, no. 2, pp. 179–187, 1962.

[82] J. Yang, S. Xie, S. Yoon, D. Park, Z. Fang, and S. Yang, "Fingerprint matching based on extreme learning machine," *Neural Comput. Appl.*, vol. 22, no. 3-4, pp. 435–445, 2013.

[83] X. Xie, F. Su, and A. Cai, "Ridge-based fingerprint recognition," in *Advances in Biometrics*. Springer, 2005, pp. 273–279.

[84] J. Feng, Z. Ouyang, and A. Cai, "Fingerprint matching using ridges," *Pattern Recognit.*, vol. 39, no. 11, pp. 2131–2140, 2006.

[85] U. Park, S. Pankanti, and A. Jain, "Fingerprint verification using SIFT features," in *SPIE Defense and Security Symposium*. International Society for Optics and Photonics, 2008, pp. 69 440K–69 440K.

[86] J. V. Kulkarni, B. D. Patil, and R. S. Holambe, "Orientation feature for fingerprint matching," *Pattern Recognit.*, vol. 39, no. 8, pp. 1551–1554, 2006.

[87] J. Li, W.-Y. Yau, and H. Wang, "Combining singular points and orientation image information for fingerprint classification," *Pattern Recognit.*, vol. 41, no. 1, pp. 353–366, 2008.

[88] J. Kour and N. Awasthy, "Nonminutiae based fingerprint matching," in *Computer Science and Information Technology-Spring Conference, 2009. IACSITSC'09. International Association of*. IEEE, 2009, pp. 199–203.

[89] J. D. Stosz and L. A. Alyea, "Automated system for fingerprint authentication using pores and ridge structure," in *SPIE's 1994 International Symposium on Optics,*

Imaging, and Instrumentation. International Society for Optics and Photonics, 1994, pp. 210–223.

[90] Q. Zhao, L. Zhang, D. Zhang, and N. Luo, "Direct pore matching for fingerprint recognition," in *Advances in Biometrics*. Springer, 2009, pp. 597–606.

[91] V. L. Jothi and S. Arumugam, "Agglomerative multi-clustering process for fingerprint matching using level 3 features." *Aust. J. Basic Appl. Sci.*, vol. 7, no. 14, 2013.

[92] A. K. Jain, K. Nandakumar, and A. Nagar, "Biometric template security," *EURASIP J. Adv. Signal Process*, vol. 2008, pp. 1–17, Jan. 2008.

[93] C. Rathgeb and A. Uhl, "A survey on biometric cryptosystems and cancelable biometrics," *EURASIP J. Inf. Secur.*, vol. 2011, no. 1, pp. 1–25, 2011.

[94] R. Ang, R. Safavi-Naini, and L. McAven, "Cancelable key-based fingerprint template," in *Proceedings of 10th Australasian Conference (ACISP '05) Information Security and Privacy*, ser. Lecture Notes in Computer Science, 2005, pp. 109–128.

[95] Y. Sutcu, H. T. Sencar, and N. Memon, "A geometric transformation to protect minutiae-based fingerprint templates," in *Proc. SPIE*, ser. Biometric Technology for Human Identification IV, vol. 6539, June 2007.

[96] H. Yang, X. Jiang, and A. C. Kot, "Generating secure cancelable fingerprint templates using local and global features," in *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*. IEEE, 2009, pp. 645–649.

[97] S. Tulyakov, F. Farooq, P. Mansukhani, and V. Govindaraju, "Symmteric hash functions for secure fingerprint biometric systems," *Pattern Recognit. Lett.*, vol. 28, pp. 2427–2436, 2007.

[98] G. Kumar, S. Tulyakov, and V. Gavindaraju, "Combination of symmetric hash functions for secure fingerprint matching," in *20th International Conference on Pattern Recognition*, Istanbul, 2010, pp. 890–893.

[99] C. Lee, J.-Y. Choi, K.-A. Toh, S. Lee, and J. Kim, "Alignment-free cancelable fingerprint templates based on local minutiae information," *IEEE Trans. Syst., Man, Cybern. B*, vol. 37, pp. 980–992, 2007.

[100] F. Farooq, R. M. Bolle, T.-Y. Jea, and N. Ratha, "Anonymous and revocable fingerprint recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, 2007, pp. 1–7.

[101] N. K. Ratha, S. Chikkerur, J. H. Connell, and R. M. Bolle, "Generating cancelable fingerprint templates," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 561–572, 2007.

[102] 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*, vol. 2, Beijing, 2008, pp. 572–575.

[103] M. Ferrara, D. Maltoni, and R. Cappelli, "Noninvertible minutia cylinder-code representation," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 6, pp. 1727–1737, 2012.

[104] N. Zhang, X. Yang, Y. Zang, X. Jia, and J. Tian, "Generating registration-free cancelable fingerprint templates based on minutia cylinder-code representation," in *Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE Sixth International Conference on*. IEEE, 2013, pp. 1–6.

[105] W. Yang, J. Hu, S. Wang, and J. Yang, "Cancelable fingerprint templates with Delaunay triangle-based local structures," in *Cyberspace Safety and Security*. Springer, 2013, pp. 81–91.

[106] S. Wang and J. Hu, "Design of alignment-free cancelable fingerprint templates via curtailed circular convolution," *Pattern Recognit.*, vol. 47, no. 3, pp. 1321–1329, 2014.

[107] Y. Sutcu, S. Rane, J. S. Yedidia, S. C. Draper, and A. Vetro, "Feature extraction for a Slepian-Wolf biometric system using LDPC codes," in *Information Theory, 2008 (ISIT '08). IEEE International Symposium on*, Toronto, Canada, July 2008, pp. 2297–2301.

[108] A. Nagar, S. Rane, and A. Vetro, "Privacy and security of features extracted from minutiae aggregates," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, Dallas, TX, March 2010, pp. 1826–1829.

[109] T. Ahmad, J. Hu, and S. Wang, "String-based cancelable fingerprint templates," in *Industrial Electronics and Applications (ICIEA), 2011 6th IEEE Conference on*, Beijing, June 2011, pp. 1028–1033.

[110] A. B. J. Teoh, D. C. L. Ngo, and A. Goh, "Biohashing: two factor authentication featuring fingerprint data and tokenised random number," *Pattern Recognit.*, vol. 37, pp. 2245–2255, 2004.

[111] A. B. J. Teoh, W. K. Yip, and S. Lee, "Cancellable biometrics and annotations on biohash," *Pattern Recognit.*, vol. 41, pp. 2034–2044, 2008.

[112] A. B. J. Teoh, W. K. Yip, and K.-A. Toh, "Cancellable biometrics and user-dependent multi-state discretization in biohash," *Pattern Anal. Appl.*, vol. 13, pp. 301–307, 2010.

[113] S. Wang and J. Hu, "Alignment-free cancelable fingerprint template design: A densely infinite-to-one mapping (DITOM) approach," *Pattern Recognit.*, vol. 45, no. 12, pp. 4129–4137, 2012.

[114] 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 Recognit. Lett.*, vol. 42, pp. 137–147, 2014.

[115] Z. Jin and A. B. J. Teoh, "Fingerprint template protection with minutia vicinity decomposition," in *Biometrics (IJCB), 2011 International Joint Conference on*. IEEE, 2011, pp. 1–7.

[116] C. Lee and J. Kim, "Cancelable fingerprint templates using minutiae-based bit-strings," *J. Netw. Comput. Appl.*, vol. 33, pp. 236–246, 2010.

[117] Z. Jin, T. S. Ong, C. Tee, and A. B. J. Teoh, "Generating revocable fingerprint template using polar grid based 3-tuple quantization technique," in *IEEE 54th International Midwest Symposium on Circuits and Systems*, Seoul, 2011, pp. 1–4.

[118] Z. Jin, A. B. J. Teoh, T. S. Ong, and C. Tee, "Generating revocable fingerprint template using minutiae pair representation," in *2nd International Conference on Education Technology and Computer*, Shanghai, 2010, pp. 251–255.

[119] B. Yang, D. Hartung, K. Simoens, and C. Busch, "Dynamic random projection for biometric template protection," in *Biometrics: Theory, apllications and systems (BTAS), Fourth IEEE International Conference on*, Washington, DC, 2010, pp. 1–7.

[120] K. Takahashi and S. Hirata, "Generating provably secure cancelable fingerprint templates based on correlation-invariant random filtering," in *2011 International Joint Conference on Biometrics (IJCB)*, Piscataway, NJ, 2011, pp. 1–8.

[121] M. Mimura, S. Ishida, and Y. Seto, "Development of personal authentication techniques using fingerprint matching embedded in smart cards," *IECIE Trans. Inf. Syst.*, vol. E84-D, no. 7, pp. 812–818, 2001.

[122] K. Takahashi, "Unconditionally provably secure cancelable biometrics based on a quotient polynomial ring," in *the 3rd IEEE International Conference on Biometrics: Theory, applications and systems*, Washington, DC, 2009, pp. 327–332.

[123] R. Belguechi, M. E. A. Cherrier, and C. Rosenberger, "Biohashing for securing fingerprint minutiae templates," in *the 20th International Conference on Pattern Recognition (ICPR)*, Istanbul, 2010, pp. 1168–1171.

[124] R. Belguechi, E. Cherrier, C. Rosenberger, and S. Ait-Aoudia, "An integrated framework combining Bio-Hashed minutiae template and PKCS15 compliant card for a better secure management of fingerprint cancelable templates," *Comput. Secur.*, vol. 39, pp. 325–339, 2013.

[125] ——, "Operational bio-hash to preserve privacy of fingerprint minutiae templates," *IET Biom.*, vol. 2, no. 2, pp. 76–84, 2013.

[126] A. B. J. Teoh and D. C. L. Ngo, "Biophasor: Token supplemented cancellable biometrics," in *Control, Automation, Robotics and Vision, 2006. ICARCV'06. 9th International Conference on*. IEEE, 2006, pp. 1–5.

[127] A. Arakala, J. Jeffers, and K. Horadam, "Fuzzy extractors for minutiae-based fingerprint authentication," in *Advances in Biometrics*. Springer, 2007, pp. 760–769.

[128] A. Arakala, K. Horadam, J. Jeffers, and S. Boztaş, "Protection of minutiae-based templates using biocryptographic constructs in the set difference metric," *Secur. Commun. Netw.*, vol. 4, no. 5, pp. 563–576, 2011.

[129] E.-C. Chang and S. Roy, "Robust extraction of secret bits from minutiae," in *Advances in Biometrics*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, vol. 4642, pp. 750–759.

[130] Q. Li, M. Guo, and E.-C. Chang, "Fuzzy extractors for asymmetric biometric representations," in *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*. IEEE, 2008, pp. 1–6.

[131] E. Liu, J. Liang, L. Pang, M. Xie, and J. Tian, "Minutiae and modified Biocode fusion for fingerprint-based key generation," *J. Netw. Comput. Appl.*, vol. 33, no. 3, pp. 221–235, 2010.

[132] R. Cramer, Y. Dodis, S. Fehr, C. Padró, and D. Wichs, "Detection of algebraic manipulation with applications to robust secret sharing and fuzzy extractors," in *Advances in Cryptology–EUROCRYPT 2008*. Springer, 2008, pp. 471–488.

[133] Y. Bo, S. Aidong, and Z. Wenzheng, "A fully robust fuzzy extractor," in *Cyber-Enabled Distributed Computing and Knowledge Discovery, 2009. CyberC'09. International Conference on*. IEEE, 2009, pp. 392–395.

[134] V. V. T. Tong, H. Sibert, J. Lecoeur, and M. Girault, "Biometric fuzzy extractors made practical: a proposal based on fingercodes," in *Advances in Biometrics*. Springer, 2007, pp. 604–613.

[135] K. Nandakumar, "A fingerprint cryptosystem based on minutiae phase spectrum," in *Information Forensics and Security (WIFS '10), 2010 IEEE International Workshop on*, Seattle, WA, December 2010, pp. 1–6.

[136] H. Xu, R. N. J. Veldhius, T. A. M. Kevenaar, A. H. M. Akkermans, and A. M. Bazen, "Spectral minutiae: A fixed-length representation for a minutiae set," in *Computer Vision and Pattern Recognition Workshops, 2008 (CVPRW '08). IEEE Computer Society Conference on*, Anchorage, AK, June 2010, pp. 1–6.

[137] K. Nandakumar, "Fingerprint matching based on minutiae phase spectrum," in *Biometrics (ICB), 5th IAPR International Conference on*, New Delhi, March-April 2012, pp. 216–221.

[138] A. Juels and M. Sudan, "A fuzzy vault scheme," *Des. Codes Cryptogr.*, vol. 38, no. 2, pp. 237–257, 2006.

[139] T. C. Clancy, N. Kiyavash, and D. J. Lin, "Secure smartcardbased fingerprint authentication," in *Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications*. ACM, 2003, pp. 45–52.

[140] U. Uludag, S. Pankanti, and A. K. Jain, "Fuzzy vault for fingerprints," in *Audio- and Video-Based Biometric Person Authentication*. Springer, 2005, pp. 310–319.

[141] U. Uludag and A. K. Jain, "Fuzzy fingerprint vault," in *Proc. Workshop: Biometrics: Challenges Arising from Theory to Practice*, 2004, pp. 13–16.

[142] S. Yang and I. Verbauwhede, "Automatic secure fingerprint verification system based on fuzzy vault scheme," in *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*, vol. 5. IEEE, 2005, pp. v–609.

[143] K. Nandakumar, A. K. Jain, and S. Pankanti, "Fingerprint-based fuzzy vault: Implementation and performance," *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 4, pp. 744–757, 2007.

[144] A. Nagar and S. Chaudhury, "Biometrics based asymmetric cryptosystem design using modified fuzzy vault scheme," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 4. IEEE, 2006, pp. 537–540.

[145] Y. Chung, D. Moon, S. Lee, S. Jung, T. Kim, and D. Ahn, "Automatic alignment of fingerprint features for fuzzy fingerprint vault," in *Information Security and Cryptology*. Springer, 2005, pp. 358–369.

[146] J. Jeffers and A. Arakala, "Minutiae-based structures for a fuzzy vault," in *Biometric Consortium Conference, 2006 Biometrics Symposium: Special Session on Research at the*. IEEE, 2006, pp. 1–6.

[147] V. Krivokuća and W. Abdulla, "Fast fingerprint alignment method based on minutiae orientation histograms," in *Proceedings of the 27th Conference on Image and Vision Computing New Zealand*. ACM, 2012, pp. 486–491.

[148] P. Li, X. Yang, K. Cao, X. Tao, R. Wang, and J. Tian, "An alignment-free fingerprint cryptosystem based on fuzzy vault scheme," *J. Netw. Comput. Appl.*, vol. 33, no. 3, pp. 207–220, 2010.

[149] K. Nandakumar, A. Nagar, and A. K. Jain, "Hardening fingerprint fuzzy vault using password," in *Advances in biometrics*. Springer, 2007, pp. 927–937.

[150] A. Nagar, K. Nandakumar, and A. K. Jain, "Securing fingerprint template: Fuzzy vault with minutiae descriptors," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*. IEEE, 2008, pp. 1–4.

[151] ——, "A hybrid biometric cryptosystem for securing fingerprint minutiae templates," *Pattern Recognit. Lett.*, vol. 31, no. 8, pp. 733–741, 2010.

[152] D. Moon, W.-Y. Choi, K. Moon, and Y. Chung, "Fuzzy fingerprint vault using multiple polynomials," in *Consumer Electronics, 2009. ISCE'09. IEEE 13th International Symposium on*. IEEE, 2009, pp. 290–293.

[153] A. Nagar, K. Nandakumar, and A. K. Jain, "Multibiometric cryptosystems based on feature-level fusion," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 1, pp. 255–268, 2012.

[154] T. H. Nguyen, Y. Wang, Y. Ha, and R. Li, "Improved chaff point generation for vault scheme in bio-cryptosystems," *IET Biom.*, vol. 2, no. 2, pp. 48–55, 2013.

[155] Q. FENG, Y.-y. XIAO, F. SU, and A.-n. CAI, "Cancelable fingerprint fuzzy vault scheme," *J. Comput. Appl.*, vol. 7, p. 060, 2008.

[156] D. Xu and X. Wang, "A scheme for cancelable fingerprint fuzzy vault based on chaotic sequence," in *Mechatronics and Automation (ICMA), 2010 International Conference on*. IEEE, 2010, pp. 329–332.

[157] J. Bringer, H. Chabanne, and B. Kindarji, "The best of both worlds: Applying secure sketches to cancelable biometrics," *Sci. Comput. Program.*, vol. 74, no. 1, pp. 43–51, 2008.

[158] P. Tuyls, A. H. M. Akkermans, T. A. M. Kevenaar, G.-J. Schrijen, A. M. Bazen, and R. N. J. Veldhuis, "Practical biometric authentication with template protection," in *Audio- and Video-Based Biometric Person Authentication*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2005, vol. 3546, pp. 436–446.

[159] J. Bringer, H. Chabanne, G. Cohen, B. Kindarji, and G. Zémor, "Optimal iris fuzzy sketches," in *Biometrics: Theory, Applications, and Systems, 2007. BTAS 2007. First IEEE International Conference on*. IEEE, 2007, pp. 1–6.

[160] S. Kanade, D. Camara, E. Krichen, D. Petrovska-DelacreÌĄtaz, and B. Dorizzi, "Three factor scheme for biometric-based cryptographic key regeneration using iris," in *Biometrics Symposium, 2008. BSYM'08*. IEEE, 2008, pp. 59–64.

[161] S. Kanade, D. Petrovska-Delacrétaz, and B. Dorizzi, "Generating and sharing biometrics based session keys for secure cryptographic applications," in *Biometrics:*

Theory Applications and Systems (BTAS), 2010 Fourth IEEE International Conference on. IEEE, 2010, pp. 1–7.

[162] M. Fouad, A. El Saddik, J. Zhao, and E. Petriu, "A fuzzy vault implementation for securing revocable iris templates," in *Systems Conference (SysCon), 2011 IEEE International*. IEEE, 2011, pp. 491–494.

[163] Y. C. Feng, P. C. Yuen, and A. K. Jain, "A hybrid approach for generating secure and discriminating face template," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 1, pp. 103–117, 2010.

[164] L. Leng and J. Zhang, "Dual-key-binding cancelable palmprint cryptosystem for palmprint protection and information security," *J. Netw. Comput. Applications.*, vol. 34, no. 6, pp. 1979–1989, 2011.

[165] M. Jeong, C. Lee, J. Kim, J.-Y. Choi, K.-A. Toh, and J. Kim, "Changeable biometrics for appearance based face recognition," in *Biometric Consortium Conference, 2006 Biometrics Symposium: Special Session on Research at the*. IEEE, 2006, pp. 1–5.

[166] M. Jeong, J.-Y. Choi, and J. Kim, "Using genetic algorithms to improve matching performance of changeable biometrics from combining PCA and ICA methods," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–5.

[167] E. J. Kelkboom, J. Breebaart, I. Buhan, and R. N. Veldhuis, "Analytical template protection performance and maximum key size given a Gaussian-modeled biometric source," in *SPIE Defense, Security, and Sensing*. International Society for Optics and Photonics, 2010, pp. 76 670D–76 670D.

[168] W. J. Wong, M. L. D. Wong, and Y. H. Kho, "A low complexity multi-line code for cancelable fingerprint template," in *the 2nd International Conference on Convergence Technology 2012*, vol. 1, no. 2. Qingdao, China: Korea Convergence Society, July 2012, pp. 61–65.

[169] W.-J. Wong, M.-L. D. Wong, and Y.-H. Kho, "Multi-line code: A low complexity revocable fingerprint template for cancelable biometrics," *J. Cent. South Univ.*, vol. 20, no. 5, pp. 1292–1297, 2013.

[170] "Fingerprint Verficiation Competition FVC 2002," http://bias.csr.unibo.it/fvc2002.

[171] "Fingerprint Verficiation Competition FVC 2004," http://bias.csr.unibo.it/fvc2004.

[172] Neurotechnology, "VeriFinger SDK," Vilnius, Lithuania, 2014, version 7.0.

[173] R. Thai, "Fingerprint image enhancement and minutiae extraction," *The University of Western Australia*, 2003.

[174] S. Dasgupta, "Learning mixtures of Gaussians," in *Foundations of Computer Science, 1999. 40th Annual Symposium on*. IEEE, 1999, pp. 634–644.

[175] E. Bingham and H. Mannila, "Random projection in dimensionality reduction: applications to image and text data," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2001, pp. 245–250.

[176] Z. Jin, B.-M. Goi, A. Teoh, and Y. H. Tay, "A two-dimensional random projected minutiae vicinity decomposition-based cancellable fingerprint template," *Secur. Commun. Netw.*, 2013.

[177] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[178] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, 1993.

[179] Y. C. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*. IEEE, 1993, pp. 40–44.

[180] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1998.

[181] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[182] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley & Sons, 1991, ch. 9, pp. 224–238.

[183] B. Bhanu and X. Tan, "Fingerprint indexing based on novel features of minutiae triplets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 616–622, 2003.

[184] R. Cappelli, M. Ferrara, and D. Maltoni, "Fingerprint indexing based on minutia cylinder-code," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 1051–1057, 2011.

[185] A. Lumini, D. Maio, and D. Maltoni, "Continuous versus exclusive classification for fingerprint retrieval," *Pattern Recognit. Lett.*, vol. 18, no. 10, pp. 1027–1034, 1997.

[186] M. Scholz, F. Kaplan, C. L. Guy, J. Kopka, and J. Selbig, "Non-linear PCA: a missing data approach," *Bioinformatics*, vol. 21, no. 20, pp. 3887–3895, 2005.

[187] B. Schölkopf, A. Smola, and K.-R. Müller, "Kernel principal component analysis," in *Artificial Neural Networks âĂŤ ICANN'97*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 1997, vol. 1327, pp. 583–588.

[188] H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," *J. Comput. Graph. Stat.*, vol. 15, no. 2, pp. 265–286, 2006.

[189] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cogn. Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.

[190] K. I. Kim, K. Jung, and H. J. Kim, "Face recognition using kernel principal component analysis," *IEEE Signal Process. Lett.*, vol. 9, no. 2, pp. 40–42, 2002.

[191] J. Yang, D. Zhang, A. F. Frangi, and J.-y. Yang, "Two-dimensional PCA: a new approach to appearance-based face representation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 131–137, 2004.

[192] G. Lu, D. Zhang, and K. Wang, "Palmprint recognition using eigenpalms features," *Pattern Recognit. Lett.*, vol. 24, no. 9, pp. 1463–1467, 2003.

[193] P. N. Belhumeur, J. P. Hespanha, and D. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," vol. 19, no. 7, pp. 711–720, 1997.

[194] Q. Liu, R. Huang, H. Lu, and S. Ma, "Face recognition using kernel-based fisher discriminant analysis," in *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*. IEEE, 2002, pp. 197–201.

[195] W. J. Wong, A. B. J. Teoh, Y. H. Kho, and M. L. D. Wong, "Kernel PCA enabled bit-string representation for minutiae-based cancellable fingerprint template," *Pattern Recognit.*, vol. 51, pp. 197 – 208, 2016.

[196] V. Y. Gudkov and O. Ushmaev, "A topologic approach to user-dependent key extraction from fingerprints," in *Pattern Recognition (ICPR), 20th International Conference on*, Istanbul, August 2010, pp. 1281–1284.

[197] A. Vij and A. Namboodiri, "Learning minutiae neighborhoods: A new binary representation for matching fingerprints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 64–69.

[198] Y. Luo, J. Feng, and J. Zhou, "Fingerprint matching based on global minutia cylinder code," in *Biometrics (IJCB), 2014 IEEE International Joint Conference on*. IEEE, 2014, pp. 1–8.

[199] K. Rieck, C. Wressnegger, and A. Bikadorov, "Sally: a tool for embedding strings in vector spaces," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 3247–3251, 2012.

[200] B. Spillmann, M. Neuhaus, H. Bunke, E. Pękalska, and R. P. Duin, "Transforming strings to vector spaces using prototype selection," in *Structural, Syntactic, and Statistical Pattern Recognition*. Springer, 2006, pp. 287–296.

[201] S. Sonnenburg, A. Zien, and G. Rätsch, "Arts: accurate recognition of transcription starts in human," *Bioinform.*, vol. 22, no. 14, pp. e472–e480, 2006.

[202] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, no. 5, pp. 1299–1319, 1998.

[203] H. Hoffmann, "Kernel PCA for novelty detection," *Pattern Recognit.*, vol. 40, no. 3, pp. 863 – 874, 2007.

[204] S. Romdhani, S. Gong, and A. Psarrou, "A multi-view nonlinear active shape model using kernel PCA," in *Proc. BMVC*, 1999, pp. 48.1–48.10.

[205] S. W. Choi, C. Lee, J.-M. Lee, J. H. Park, and I.-B. Lee, "Fault detection and identification of nonlinear processes based on kernel PCA," *Chemom. Intell. Lab. Syst.*, vol. 75, no. 1, pp. 55–67, 2005.

[206] J. T. Kwok and H. Zhao, "Incremental eigen decomposition," *Matrix*, vol. 100, no. C1, p. C2, 2003.

[207] M. Artač, M. Jogan, and A. Leonardis, "Incremental PCA for on-line visual learning and recognition," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, vol. 3. IEEE, 2002, pp. 781–784.

[208] F. Riesz and B. S. Nagy, *Functional Analysis*. New York: Federick Ungar, 1955.

[209] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Data Min. Knowl. Discov.*, vol. 2, no. 2, pp. 121–167, 1998.

[210] P. F. Evangelista, M. J. Embrechts, and B. K. Szymanski, "Some properties of the gaussian kernel for one class learning," in *Artificial Neural Networks–ICANN 2007*. Springer, 2007, pp. 269–278.

[211] N. Cliff, "The eigenvalues-greater-than-one rule and the reliability of components." *Psychol. Bull.*, vol. 103, no. 2, p. 276, 1988.

[212] L. Guttman, "Some necessary conditions for common-factor analysis," *Psychometrika*, vol. 19, no. 2, pp. 149–161, 1954.

[213] H. F. Kaiser, "The application of electronic computers to factor analysis." *Educ. Psychol. Meas.*, 1960.

[214] F. B. Bryant and P. R. Yarnold, "Principal-components analysis and exploratory and confirmatory factor analysis." in *Reading and Understanding Multivariate Statistics*. American Psychological Association, 1995.

[215] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 1470–1477.

[216] Y.-G. Jiang, C.-W. Ngo, and J. Yang, "Towards optimal bag-of-features for object categorization and semantic video retrieval," in *Proceedings of the 6th ACM international conference on Image and video retrieval*. ACM, 2007, pp. 494–501.

[217] W. J. Wong, M. L. D. Wong, Y. H. Kho, and A. B. J. Teoh, "Minutiae set to bit-string conversion using multi-scale bag-of-words paradigm," in *Information Forensics and Security (WIFS), 2014 IEEE International Workshop on*, Atlanta, GA, December 2014, pp. 1–6.

[218] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, 2006.

[219] R. Rubinstein, A. M. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *Proc. IEEE*, vol. 98, no. 6, pp. 1045–1057, 2010.

[220] R. Mazhar and P. D. Gader, "EK-SVD: Optimized dictionary design for sparse representations," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*. IEEE, 2008, pp. 1–4.

[221] Z. Jiang, Z. Lin, and L. S. Davis, "Label consistent K-SVD: learning a discriminative dictionary for recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2651–2664, 2013.

[222] Y. Zhou, H. Zhao, L. Shang, and T. Liu, "Immune K-SVD algorithm for dictionary learning in speech denoising," *Neurocomputing*, vol. 137, pp. 223–233, 2014.

[223] F. Bianconi and A. Fernández, "A unifying framework for LBP and related methods," in *Local Binary Patterns: New Variants and Applications*. Springer, 2014, pp. 17–46.

[224] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[225] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 2559–2566.

[226] R. Fuksis, A. Kadikis, and M. Greitans, "Biohashing and fusion of palmprint and palm vein biometric data," in *Hand-Based Biometrics (ICHB '11), 2011 International Conference on*, Hong Kong, November 2012, pp. 1–6.

[227] C. Vielhauer, R. Steinmetz, and A. Mayerhöfer, "Biometric hash based on statistical features of online signatures," in *Proceedings of the 16th International Conference on Pattern Recognition (ICPR '02)*, vol. 1, Quebec, Canada, August 2002, pp. 123–126.

[228] H. Feng and C. C. Wah, "Private key generation from on-line handwritten signatures," *Information Management & Computer Security*, vol. 10, no. 4, pp. 159–164, 2002.

[229] Y.-J. Chang, W. Zhang, and T. Chen, "Biometrics-based cryptographic key generation," in *Multimedia and Expo (ICME '04), 2004 International Conference on*, vol. 4, June 2004, pp. 2203–2206.

[230] A. B. J. Teoh, K.-A. Toh, and W. K. Yip, "2^N discretisation of biophasor in cancellable biometrics," in *Advances in Biometrics*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, vol. 4642, pp. 435–444.

[231] M.-H. Lim, A. B. J. Teoh, and K.-A. Toh, "An efficient dynamic reliability-dependent bit allocation for biometric discretization," *Pattern Recognit.*, vol. 45, no. 5, pp. 1960–1971, 2012.

[232] C. Chen, R. N. J. Veldhuis, T. A. M. Kevenaar, and A. H. M. Akkermans, "Biometric quantization through detection rate optimized bit allocation," *EURASIP J. Adv. Signal Process.*, vol. 2009, no. 29, pp. 1–16, 2009.

[233] C. Chen and R. Veldhuis, "Extracting biometric binary strings with minimal area under the frr curve for the hamming distance classifier," *Signal Process.*, vol. 91, no. 4, pp. 906–918, 2011.

[234] X. Zhou, A. Kuijper, R. Veldhuis, and C. Busch, "Quantifying privacy and security of biometric fuzzy commitment," in *Biometrics (IJCB), 2011 International Joint Conference on*. IEEE, 2011, pp. 1–8.

[235] C. Chow and C. Liu, "Approximating discrete probability distributions with dependence trees," *IEEE Trans. Inf. Theory*, vol. 14, no. 3, pp. 462–467, 1968.

[236] M. Gomez-Barrero, J. Galbally, J. Fierrez, and J. Ortega-Garcia, "Face verification put to test: A hill-climbing attack based on the uphill-simplex algorithm," in *Biometrics (ICB), 2012 5th IAPR International Conference on*. IEEE, 2012, pp. 40–45.

[237] E. Maiorana, G. E. Hine, and P. Campisi, "Hill-climbing attacks on multibiometrics recognition systems," 2013.

[238] W. J. Wong, M. L. D. Wong, and A. B. J. Teoh, "A security- and privacy-driven hybrid biometric template protection technique," in *Electronics, Information and Communications (ICEIC), 2014 International Conference on*, Kota Kinabalu, Malaysia, January 2014, pp. 1–5.

[239] R. W. Hamming, "Error detecting and error correcting codes," *Bell System Technical Journal*, vol. 29, no. 2, pp. 147–160, 1950.

[240] R. J. McEliece, "The reliability of computer memories," *Scientific American*, vol. 252, no. 1, pp. 88–95, 1985.

[241] E. R. Berlekamp, *Nonbinary BCH decoding*. University of North Carolina. Department of Statistics, 1966.

[242] R. T. Chien, "Cyclic decoding procedures for bose-chaudhuri-hocquenghem codes." *IEEE Trans. Inf. Theory*, vol. 10, no. 4, pp. 357–363, 1964.

[243] T. H. Nguyen, Y. Wang, Y. Ha, and R. Li, "Performance and security-enhanced fuzzy vault scheme based on ridge features for distorted fingerprints," *IET Biometrics*, vol. 4, no. 1, pp. 29–39, 2015.

[244] W. Yang, J. Hu, S. Wang, and M. Stojmenovic, "An alignment-free fingerprint biocryptosystem based on modified voronoi neighbor structures," *Pattern Recognit.*, vol. 47, no. 3, pp. 1309–1320, 2014.

[245] U. Uludag and A. Jain, "Securing fingerprint template: Fuzzy vault with helper data," in *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on*. IEEE, 2006, pp. 163–163.

[246] C. R. Rao, *Linear statistical inference and its applications*. John Wiley & Sons, 2009, vol. 22.

[247] K. Nandakumar and A. K. Jain, "Biometric template protection: Bridging the performance gap between theory and practice," *IEEE Signal Processing Magazine - Special Issue on Biometric Security and Privacy*, Sep 2015.

[248] W. Barbakh and C. Fyfe, "Online clustering algorithms," *International J. Neural Syst.*, vol. 18, no. 03, pp. 185–194, 2008.

[249] A. Choromanska and C. Monteleoni, "Online clustering with experts," in *International Conference on Artificial Intelligence and Statistics*, 2012, pp. 227–235.

[250] N. Ailon, R. Jaiswal, and C. Monteleoni, "Streaming k-means approximation," in *Advances in Neural Information Processing Systems*, 2009, pp. 10–18.

[251] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 689–696.

[252] C. Lu, J. Shi, and J. Jia, "Online robust dictionary learning," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 415–422.

[253] L. Fisher and J. W. Van Ness, "Admissible clustering procedures," *Biometrika*, vol. 58, no. 1, pp. 91–104, 1971.

[254] M. V. Prasad and C. S. Kumar, "Fingerprint template protection using multiline neighboring relation," *Expert Syst. with Appl.*, vol. 41, no. 14, pp. 6114–6122, 2014.

[255] N. Abe and T. Shinzaki, "Vectorized fingerprint representation using minutiae relation code," in *Biometrics (ICB), 2015 International Conference on*. IEEE, 2015, pp. 408–415.

[256] J. S. Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. New York: Cambridge University Press, 2004.

Appendices

APPENDIX A

Gram-Schmidt Orthogonalization

Gram-Schmidt orthogonalization, also known as the Gram-Schmidt process, is a technique which takes non-orthogonal and linearly independent set $S = \{\mathbf{v}_1, ..., \mathbf{v}_k\}$ and generates an orthogonal set $S\prime = \{\mathbf{u}_1, ..., \mathbf{u}_k\}$. Let

$$\text{proj}_{\mathbf{u}}(\mathbf{v}) := \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\langle \mathbf{u}, \mathbf{u} \rangle} \mathbf{u}, \tag{A.0.1}$$

where $\langle \mathbf{u}, \mathbf{v} \rangle$ denotes the inner product of the vectors $\mathbf{u}$ and $\mathbf{v}$. The output vectors of the orthogonal set are computed as follows:

$$\begin{aligned}
\mathbf{u}_1 &= \mathbf{v}_1, & \mathbf{e}_1 &= \frac{\mathbf{u}_1}{\|\mathbf{u}_1\|}; \\
\mathbf{u}_2 &= \mathbf{v}_2 - \text{proj}_{\mathbf{u}_1}(\mathbf{v}_2), & \mathbf{e}_2 &= \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|}; \\
\mathbf{u}_3 &= \mathbf{v}_3 - \text{proj}_{\mathbf{u}_1}(\mathbf{v}_3) - \text{proj}_{\mathbf{u}_2}(\mathbf{v}_3), & \mathbf{e}_3 &= \frac{\mathbf{u}_3}{\|\mathbf{u}_3\|}; \\
&\vdots & & \\
\mathbf{u}_k &= \mathbf{v}_k - \sum_{j=1}^{k-1} \text{proj}_{\mathbf{u}_j}(\mathbf{v}_k), & \mathbf{e}_k &= \frac{\mathbf{u}_k}{\|\mathbf{u}_k\|},
\end{aligned} \tag{A.0.2}$$

where $\|\cdot\|$ denotes the ℓ^2 norm. The set $\{\mathbf{e}_1, ..., \mathbf{e}_k\}$ contains normalized vectors of $\{\mathbf{u}_1, ..., \mathbf{u}_k\}$, hence the process of calculating $\{\mathbf{e}_1, ..., \mathbf{e}_k\}$ is also called Gram-Schmidt orthonormalization.

APPENDIX B

Kernel Validation(Mercer's Theorem)

Theorem 1 (Mercer's Theorem [208]). *Any continuous, symmetric and semi-definite function $k(x, y)$ can be used as a kernel function if and only if, for any $g(x)$ such that*

$$\int [g(x)]^2 dx < \infty \tag{B.0.1}$$

then

$$\iint k(x, y)g(x)g(y)dxdy \geq 0 \tag{B.0.2}$$

Lemma 1 ([256]). *Let $k_1(x, y)$ and $k_2(x, y)$ be kernels over the data space $X \times X$, $X \subseteq \mathbb{R}^n$ and $a \in \mathbb{R}^+$, then the following functions are kernels which satisfy the Mercer's theorem:*

1. $k(x, y) = k_1(x, y) + k_2(x, y)$
2. $k(x, y) = k_1(x, y)k_2(x, y)$
3. $k(x, y) = ak_1(x, y)$
4. $k(x, y) = \exp(k_1(x, y))$

APPENDIX C

Example of BCH Encoding

Given that α is the primitive element of a Galois field (GF) (refer to section 7.2.2 for the notions of GF), Table C.1 shows an example of the elements in GF(2^4) in both polynomial form and vector form. The polynomial representations of the elements with degree 4 or higher are generated based on the definition $\mathbf{p}(\alpha) = \alpha^4 + \alpha + 1 = 0 \Rightarrow \alpha^4 = \alpha + 1$.

Table C.1: Elements of GF(2^4) generated by the primitive polynomial $\mathbf{p}(X) = X^4 + X + 1$.

Power representation	Polynomial representation	Vector representation
0	0	[0,0,0,0]
1	1	[0,0,0,1]
α	α	[0,0,1,0]
α^2	α^2	[0,1,0,0]
α^3	α^3	[1,0,0,0]
α^4	$\alpha + 1$	[0,0,1,1]
α^5	$\alpha^2 + \alpha$	[0,1,1,0]
α^6	$\alpha^3 + \alpha^2$	[1,1,0,0]
α^7	$\alpha^3 + \alpha + 1$	[1,0,1,1]
α^8	$\alpha^2 + 1$	[0,1,0,1]
α^9	$\alpha^3 + \alpha$	[1,0,1,0]
α^{10}	$\alpha^2 + \alpha + 1$	[0,1,1,1]
α^{11}	$\alpha^3 + \alpha^2 + \alpha$	[1,1,1,0]
α^{12}	$\alpha^3 + \alpha^2 + \alpha + 1$	[1,1,1,1]
α^{13}	$\alpha^3 + \alpha^2 + 1$	[1,1,0,1]
α^{14}	$\alpha^3 + 1$	[1,0,0,1]

With this, the minimal polynomials of the elements can be calculated. Minimal polynomials ($\mathbf{\Lambda}(X)$) are polynomials that satisfies $\mathbf{\Lambda}(\alpha^j) = 0$. For example, the minimal

polynomial of α^3 can be obtained by solving the following using Table C.1:

$$
\begin{array}{rcrcrcrcrcl}
 & & & & & & & & \mathbf{\Lambda}(\alpha^3) & = & 0 \\
 & \Lambda_{12}(\alpha^3)^4 & + & \Lambda_9(\alpha^3)^3 & + & \Lambda_6(\alpha^3)^2 & + & \Lambda_3\alpha^3 & + \quad \Lambda_0 & = & 0 \\
 & \Lambda_{12}\alpha^3 & + & \Lambda_{12}\alpha^2 & + & \Lambda_{12}\alpha & + & \Lambda_{12} & & & \\
+ & \Lambda_9\alpha^3 & & & + & \Lambda_9\alpha & & & & & \\
+ & \Lambda_6\alpha^3 & + & \Lambda_6\alpha^2 & & & & & & & \\
+ & \Lambda_3\alpha^3 & & & & & & & & & \\
 & & & & & & & & + \quad \Lambda_0 & & \\
\hline
 & \cdots & + & \cdots & + & \cdots & + & \cdots & + \quad \cdots & = & 0
\end{array}
$$

$$\text{get } \Lambda_{12} = 1, \Lambda_9 = 1, \Lambda_6 = 1, \Lambda_3 = 1, \Lambda_0 = 1,$$

$$\therefore \text{the minimal polynomial of } \alpha^3 \text{ is } \mathbf{\Lambda}_{(3)}(X) = X^4 + X^3 + X^2 + X + 1.$$

Following the same approach, the minimal polynomials of the elements of GF(2^4) for $\mathbf{p}(X) = X^4 + X + 1$ are shown in Table C.2.

Table C.2: Minimal polynomials of the elements of GF(2^4) generated by the primitive polynomial $\mathbf{p}(X) = X^4 + X + 1$.

Element(s)(α^j)	Minimal polynomial($\mathbf{\Lambda}_{(j)}(X)$)
$\{\alpha, \alpha^2, \alpha^4, \alpha^8\}$	$X^4 + X + 1$
$\{\alpha^3, \alpha^6, \alpha^9, \alpha^{12}\}$	$X^4 + X^3 + X^2 + X + 1$
$\{\alpha^5, \alpha^{10}\}$	$X^2 + X + 1$
$\{\alpha^7, \alpha^{11}, \alpha^{13}, \alpha^{14}\}$	$X^4 + X^3 + 1$

If the desired error-correcting capability is $t = 3$, then the generator polynomial of the BCH encoder is the least common multiple of the first $2t$ minimal polynomials:

$$
\begin{aligned}
\mathbf{g}(X) &= \text{LCM}(\mathbf{\Lambda}_{(1)}(X), \mathbf{\Lambda}_{(2)}(X), \mathbf{\Lambda}_{(3)}(X), \mathbf{\Lambda}_{(4)}(X), \mathbf{\Lambda}_{(5)}(X), \mathbf{\Lambda}_{(6)}(X)) \\
&= \text{LCM}(\mathbf{\Lambda}_{(1)}(X), \mathbf{\Lambda}_{(3)}(X), \mathbf{\Lambda}_{(5)}(X)) \\
&= (X^4 + X + 1)(X^4 + X^3 + X^2 + X + 1)(X^2 + X + 1) \\
&= X^{10} + X^8 + X^5 + X^4 + X^2 + X + 1
\end{aligned}
$$

For a BCH(15,5,3)-code, given a message $\boldsymbol{\kappa}_s = [1,1,0,1,1]$ or $\boldsymbol{\kappa}_s(X) = X^4 + X^3 + X + 1$, the codeword ($\mathbf{E}(X)$) is computed using (7.4.2) through the following steps:

$$\begin{aligned} X^{15-5}\boldsymbol{\kappa}_s(X) &= X^{14} + X^{13} + X^{11} + X^{10} \\ \mathbf{r}(X) &= X^{10}\boldsymbol{\kappa}_s(X) \mod \mathbf{g}(X) \\ &= X^9 + X^4 + X^2 \\ \therefore \mathbf{E}(X) &= X^{10}\boldsymbol{\kappa}_s(X) + \mathbf{r}(X) \\ &= X^{14} + X^{13} + X^{11} + X^{10} + X^9 + X^4 + X^2 \\ &\text{or} \\ \mathbf{E} &= [1,1,0,1,1,1,0,0,0,0,1,0,1,0,0] \end{aligned}$$

APPENDIX D

Example of BCH Decoding

A message has been successfully encoded into a BCH(15,5,3) codeword in Appendix C. The generator polynomial, message and the codeword are as follows:

$$\begin{aligned}
\mathbf{g}(X) &= X^{10} + X^8 + X^5 + X^4 + X^2 + X + 1 \\
\boldsymbol{\kappa}_s(X) &= X^4 + X^3 + X + 1 \\
\mathbf{E}(X) &= X^{14} + X^{13} + X^{11} + X^{10} + X^9 + X^4 + X^2
\end{aligned}$$

Now lets assume the codeword is slightly distorted during transmission and the received codeword is $\mathbf{E}'(X) = X^{11} + X^{10} + X^9 + X^4 + X^2 + 1$ or $\mathbf{E}' = [0,0,0,1,1,1,0,0,0,0,1,0,1,0,1]$. First, calculate the first $2t$ coefficients of the syndrome based on (7.4.3) and Table C.1:

$$\begin{aligned}
\text{syn}_{(1)} &= \mathbf{E}'(\alpha) &&= \alpha^8 \\
\text{syn}_{(2)} &= \mathbf{E}'(\alpha^2) &&= \alpha \\
\text{syn}_{(3)} &= \mathbf{E}'(\alpha^3) &&= \alpha^2 \\
\text{syn}_{(4)} &= \mathbf{E}'(\alpha^4) &&= \alpha^2 \\
\text{syn}_{(5)} &= \mathbf{E}'(\alpha^5) &&= 0 \\
\text{syn}_{(6)} &= \mathbf{E}'(\alpha^6) &&= \alpha^4
\end{aligned}$$

After that, the Berlekamp-Massey algorithm (as described in Algorithm 7.1) is executed to determine the error locator polynomial, $\sigma(X)$. The algorithm consists of $2t$ iterations and the values of the variables in each iteration are computed in Table D.1. The final error locator polynomial obtained is $\sigma(X) = \alpha^{12}X^3 + \alpha^7X^2 + \alpha^8X + 1$. Through brute-force Chien search, the roots of $\sigma(X)$ are $\sigma(1) = \sigma(\alpha) = \sigma(\alpha^2) = 0$. Therefore, the errors are $\mathbf{e}(X) = X^{15-1} + X^{15-2} + 1 = X^{14} + X^{13} + 1$ and successful decoding is proven by $\mathbf{E}(X) = \mathbf{E}'(X) + \mathbf{e}(X)$

Table D.1: Elements of GF(2^4) generated by the primitive polynomial $\sigma(X) = \mathbf{p}(X) = X^4 + X + 1$.

i	$\Delta^{(i)}$	$\delta^{(i)}$	$L^{(i)}$	$\mathbf{B}^{(i)}(X)$	$\sigma^{(i)}(X)$
0	-	-	0	1	1
1	α^8	1	1	α^7	$\alpha^8 X + 1$
2	0	0	1	$\alpha^7 X$	$\alpha^8 X + 1$
3	α^{11}	1	2	$\alpha^{12} X + \alpha^4$	$\alpha^3 X^2 + \alpha^8 X + 1$
4	0	0	2	$\alpha^{12} X^2 + \alpha^4 X$	$\alpha^3 X^2 + \alpha^8 X + 1$
5	1	1	3	$\alpha^3 X^2 + \alpha^8 X + 1$	$\alpha^{12} X^3 + \alpha^7 X^2 + \alpha^8 X + 1$
6	0	0	3	$\alpha^3 X^3 + \alpha^8 X^2 + X$	$\alpha^{12} X^3 + \alpha^7 X^2 + \alpha^8 X + 1$

Publications Arising from this Thesis

[C] W. J. Wong, M. L. D. Wong, Y. H. Kho and A. B. J. Teoh, "Minutiae set to bit-string conversion using multi-scale bag-of-words paradigm," in *Information Forensics and Security (WIFS), 2014 IEEE International Workshop on*, Atlanta, GA, December 2014, pp. 1-6.

[C] W. J. Wong, M. L. D. Wong, and A. B. J. Teoh, "A security- and privacy-driven hybrid biometric template protection technique," in *Electronics, Information and Communications (ICEIC), 2014 International Conference on*, Kota Kinabalu, Malaysia, January 2014, pp. 1-5.

[C] W. J. Wong, M. L. D. Wong, and Y. H. Kho, "A low complexity multi-line code for cancelable fingerprint template," in *the 2nd International Conference on Convergence Technology 2012*, vol. 1, no. 2, Qingdao, China, July 2012, pp. 61-65.

[J] W. J. Wong, A. B. J. Teoh, Y. H. Kho, and M. L. D. Wong, "Kernel PCA enabled bit-string representation for minutiae-based cancellable fingerprint template," *Pattern Recognit.*, vol. 51, pp. 197-208, 2016.

[J] W. J. Wong, A. B. J. Teoh, M. L. D. Wong, and Y. H. Kho, "Enhanced multi-line code for minutiae-based fingerprint template protection," *Pattern Recognit. Lett.*, vol. 34, no. 11, pp. 1221-1229, 2013.

[J] W.-J. Wong, M.-L. D. Wong, and Y.-H. Kho, "Multi-line code: A low complexity revocable fingerprint template for cancelable biometrics," *J. Cent. South Univ.*, vol. 20, no. 5, pp. 1292-1297, 2013.