Biometric Security

Biometric Security

Edited by

David Chek Ling Ngo,
Andrew Beng Jin Teoh
and Jiankun Hu

Cambridge
Scholars
Publishing

Biometric Security

Edited by David Chek Ling Ngo, Andrew Beng Jin Teoh and Jiankun Hu

This book first published 2015

Cambridge Scholars Publishing

Lady Stephenson Library, Newcastle upon Tyne, NE6 2PA, UK

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

ISBN (10): 1-4438-7183-4
ISBN (13): 978-1-4438-7183-9

TABLE OF CONTENTS

Preface vii

Part 1. Biometric Template Protection

Chapter One 2
Cancelable Biometrics and Data Separation Schemes
Kenta Takahashi

Chapter Two 37
Minutiae-based Fingerprint Representations: Review, Privacy, Security and Cryptographic Realization
Zhe Jin, Syh-Yuan Tan, Andrew Teoh Beng Jin and Bok-Min Goi

Chapter Three 92
Palmprint Template Protection Technologies
Lu Leng

Part 2. Biometric Key and Encryption

Chapter Four 134
Biometric Discretization for Template Protection and Cryptographic Key Generation
Meng-Hui Lim

Chapter Five 165
De-Identifying Biometric Images by Decomposition and Mixing
Asem Othman and Arun Ross

Part 3. Biometric System Analysis

Chapter Six 198
BioPACE: Biometric-Protected Authentication Connection Establishment
Nicolas Buchmann, Christian Rathgeb, Roel Peeters, Harald Baier and Christoph Busch

Chapter Seven 224
Privacy and Security Assessment of Biometric Systems
Mohamad El-Abed, Patrick Lacharme and Christophe Rosenberger

Chapter Eight 255
A Generalized Framework for Privacy and Security Assessment
of Biometric Template Protection
Xuebing Zhou and Bian Yang

Part 4. Privacy-Enhanced Biometric Systems

Chapter Nine 274
Secure and Efficient Iris and Fingerprint Identification
Marina Blanton and Paolo Gasti

Chapter Ten 312
Identification Over Outsourced Biometric Data
Julien Bringer, Hervé Chabanne and Alain Patey

Chapter Eleven 351
A Collaborative Framework Design for Distributed
Biometrics-based Authentication in the Cloud
Kok-Seng Wong and Myung Ho Kim

Chapter Twelve 381
Secure Two-Party Computation and Biometric Identification
Julien Bringer, Hervé Chabanne and Alain Patey

Part 5. Other Biometric Security Technologies

Chapter Thirteen 428
Watermarked Biometrics
Fengling Han, Ron van Schyndel and Mohammed Ahmad A Alkhathami

Chapter Fourteen 459
3D Fingerprints: A Survey
Wei Zhou, Jiankun Hu, Song Wang, Ian Petersen
and Mohammed Bennamoun

PREFACE

Modern biometrics is defined as the science of using biological properties to identify individuals. Biometrics delivers an enhanced level of security by means of a “proof of property”, where the claimant presents “proofs” that directly connect with their own intrinsic physical or behavioral characteristics. Security by means of biometrics implies that the user is spared from having to remember a password, or to carry a token, and that the identity of the user is much more difficult to duplicate or share with others, owing to the uniqueness and non-repudiation nature of biometrics.

The design and deployment of a biometric system, however, obscures many pitfalls, which, when underestimated, can lead to major security risks and privacy threats. Since there exists a strong binding between the user and their identity, biometric identity theft and privacy invasion have become issues of great concern. A biometric template, once compromised, is difficult to revoke or replace; furthermore, it is rendered unusable, just as with a password. The avoidance of a database storing biometrics, or perhaps storing them to the fullest extent possible, has emerged as a preventive and defensive measure.

This book volume is a reference work containing articles on a comprehensive range of topics that discuss recent advances and discoveries in “biometric security and privacy”, a relatively new and multidisciplinary research which emerged in the late 90’s, so to address two essential problems: the privacy concerns as well as the security concerns associated with biometric systems. It compiles a total of fourteen articles, all contributed by thirty-two eminent researchers in the field, thus providing a concise and accessible coverage of not only general issues, but also providing state-of-the-art, reliable solutions, so to address these issues in five parts: (1) Biometric Template Protection, which covers cancellable biometrics and its parameter management protocol; (2) Biometric Key and Encryption, focusing on biometric key generation and visual biometric cryptography; (3) Biometric Systems Analysis, dealing with biometric system security, and privacy evaluation and assessment; (4) Privacy Enhanced Biometric Systems, covering privacy-enhanced biometric system protocol design and implementation; and (5) Other Biometric Security Technologies.

Specifically, the book is organized as follows:

Part 1

Chapter 1, “Cancelable Biometrics and Data Separation Schemes,” discusses several typical parameter management schemes for cancellable biometrics and their limitations. The chapter introduces a scheme based on server-side parameter management, in detail, so to address the usability problem, and also discusses a number of authentication protocols for this scheme. The security and usability of the schemes are also discussed and compared.

Chapter 2, “Minutiae-based Fingerprint Representations: Review, Privacy, Security and Cryptographic Realization,” presents an overview for fixed-length and variable-size minutiae-based fingerprint representations. It makes use of three methods, so to provide a case study on the generation of fingerprint representations from minutiae. An instance of cryptographic realization using minutiae-based fingerprint representation is also demonstrated.

Chapter 3, “Palmprint Template Protection Technologies,” introduces and compares the existing palmprint template protection technologies, which can be divided into three categories, namely palmprint cryptosystems, cancelable palmprint, and hybrid methods. The future outlook of these technologies is highlighted.

Part 2

Chapter 4, “Biometric Discretization for Template Protection and Cryptographic Key Generation,” reviews recent advances on quantization, as well as on feature encoding in biometric discretization. The author also presents an extensive comparative study of several state-of-the-art discretization schemes, and suggests future directions.

Chapter 5, “Biometric Privacy Using Visual Cryptography and Mixing Techniques,” explores methods that can be used to extend privacy to biometric data in the context of an operational system. The authors discuss a method based on Visual Cryptography that de-identifies a face or fingerprint image prior to storing it by decomposing the original image into two images in such a way that the original image can be revealed only when both images are simultaneously made available; further, each component image does not reveal the identity of the original image. They also discuss a method based on the concept of mixing, so to extend privacy to fingerprint images.

Part 3

Chapter 6, “BioPACE: Biometric-Protected Authentication Connection Establishment,” introduces BioPACE, a biometrics based authentication protocol. The operation mode of BioPACE is described in detail, the integration of biometric information is investigated and a security assessment is given.

Chapter 7, “Privacy and Security Assessment of Biometric Systems,” illustrates various security and privacy issues, as well as the evaluation of biometric systems. The *EvaBio tool* - an evaluation tool for the security and privacy assessment of biometric systems, is also introduced.

Chapter 8, “A Generalized Framework for Privacy and Security Assessment of Biometric Template Protection,” establishes a comprehensive evaluation framework for biometric template security and privacy. The assessment framework is composed of three components; goals identification, threat models determination, and evaluation metrics and process development. A case study on iris fuzzy commitment is demonstrated.

Part 4

Chapter 9, “Secure and Efficient Iris and Fingerprint Identification,” presents the design, security analysis, and performance of privacy-preserving identification protocols for iris codes and fingerprints. The authors also demonstrate, with certain optimizations, that such techniques are suitable for practical use on large data sets.

Chapter 10, “Identification over Outsourced Biometric Data,” introduces several protocols for outsourcing biometric data to an untrusted server while maintaining identification functionalities without compromising confidentiality of the data or privacy of the requests.

Chapter 11, “A Collaborative Framework Design for Distributed Biometrics-based Authentication in the Cloud,” outlines a privacy-preserved and security-protected solution for biometric data stored in the cloud.

Chapter 12, “Secure Two-Party Computation and Biometric Identification,” summarizes secure Two-Party Computation concepts and techniques that can be applied to privacy-preserving biometric identification.

Part 5

Chapter 13, “Biometric Watermarking,” discusses the use of biometrics in remote identity authentication services via watermarking technology. The authors showcase a case study of watermark embedding of fingerprint images based on Wong’s original algorithm, the Discrete Cosine Transform (DCT), and the Dual Tree Complex Wavelet Transform (DTCWT).

Chapter 14, “The 3D Fingerprints-A Survey,” investigates the acquisition of 3D fingerprint images, the compatibility between 3D fingerprints and 2D fingerprints, and the feature representations of 3D fingerprints. Specific recommendations for future research directions in 3D fingerprints are also provided.

The target audience for the book includes researchers, scholars, graduate students, engineers, IT practitioners and developers who are interested in security and privacy related issues in biometric systems. Also, managers of organizations with strong security needs will find this book of great value.

The editors would like to express their sincere gratitude to all distinguished contributors who make this book possible, and the group of reviewers who have offered invaluable comments to improve the quality of each and every chapter. A dedicated team at Cambridge Scholars Publishing has also assisted the editors continuously from inception to final production of the book. We thank them for their painstaking efforts in all stages of production. We gratefully acknowledge the financial support that we have received from Sunway University.

A B J Teoh, D C L Ngo and J Hu
January 15

PART 1.

BIOMETRIC TEMPLATE PROTECTION

CHAPTER ONE

CANCELABLE BIOMETRICS AND DATA SEPARATION SCHEMES

KENTA TAKAHASHI
HITACHI, LTD., YOKOHAMA RESEARCH LABORATORY,
KANAGAWA, JAPAN

Abstract

Protecting biometric information is a critical issue in biometric systems, since biometric characteristics such as fingerprints, irises, and face and vein patterns, constitute privacy information, and more importantly, they cannot be changed or revoked like passwords. To address this issue, a privacy-preserving biometric authentication scheme called *cancellable biometrics* has been studied, in which the biometric features are transformed by a kind of encryption or one-way function, and matched without restoring the original features. The transformation function is determined by a user-specific parameter, which plays a similar role to an encryption key or a salt. To secure biometric features using cancellable biometrics, the parameters must be managed separately from the transformed features.

In this chapter, firstly, several studies on cancellable biometrics are reviewed. Secondly several typical schemes for parameter management are introduced and their limitations, mainly of usability, are discussed. Subsequently, another scheme based on server-side parameter management is introduced, so to address the usability problem, and several authentication protocols for this scheme are presented. Finally, the security and usability of the schemes are discussed and compared.

Keywords: biometrics, cancellable biometrics, template protection, information security

1 Introduction

Biometric authentication technology, a technology which automatically identifies a person based on his/her physical or behavioral features, has been used for user authentication for various applications, such as physical access control and computer application login. In future, this technology is expected to be applied to remote user authentication over networks, e.g. Internet banking, e-commerce, and various cloud services. A typical remote biometric authentication system consists of an authentication server and client terminals with biometric sensors [23]. The server retains the biometric feature data associated with user IDs called templates, in a database.

However, problems emerge. The first is a security concern: Because biometric features such as fingerprint patterns are unchangeable, unlike passwords, they cannot be changed or revoked even if the templates or feature data are compromised. The second is a privacy concern: Biometric information is strongly linked to a person's identity, and hence some users have refrained from disclosing their biometric data to servers over the network.

Conventional remote biometric authentication systems have dealt with these problems by encrypting templates in the databases, and by using cryptographic communication. However, the encrypted templates must be decrypted in the server, so to perform pattern matching at the time of authentication. Thus, a skilled attacker or a malicious administrator of the server can acquire the original templates. Biometric template protection (BTP) schemes, which address these issues, have been studied for approximately a decade, and can broadly be classified into two categories; *feature transformation* and *biometric cryptosystems* [13].

The biometric cryptosystems [38], such as ones employing *fuzzy vault* (e.g. [18]), take the approach of extracting stable binary representations from noisy biometrics data (*biometric key generation*), and using it as a cryptographic key or a password. However, since most biometric key generation methods rely on error correcting code theory, the performance, i.e. false rejection rate (FRR) and false acceptance rate (FAR), of biometric cryptosystems is limited by the error-correcting capability. Generating a stable key from noisy biometric data, but culminating in a practical performance, is a major challenge in this approach.

The feature transformation approach was first proposed by Ratha, et. al. [22], named *cancellable biometrics*. Here, we label the set of BTP methods based on this approach 'cancellable biometrics'. In cancellable biometrics, biometric features are transformed and matched in the

transformed domain, directly without restoring the original feature. The transformation function is determined by a (typically user-specific) *parameter*, which may be a set of multiple parameter values. The parameter plays a similar role as an encryption key or a salt. Even if the transformed template (the *cancellable template*) or the parameter is compromised, their effect can be revoked by changing the parameter and reissuing a cancellable template via a new parameter, without changing the original biometric features. In this chapter, we firstly provide an overview of the BTP scheme and then review several studies on cancellable biometrics.

Various methods pertaining to cancellable biometrics such as [2, 25, 21, 3, 30, 31] have the potential to take advantage of sophisticated conventional matchers, with practical accuracy. In addition, several feature transformation functions are considered to have high security in the sense that it is impossible or computationally difficult to restore or guess the original template from a cancellable template without knowing the parameter. For example, transformations proposed in [30] are mathematically proven to be information-theoretically secure.

Many of these transformations including [2, 25, 7, 30] are types of encryptions where the parameter plays a key role. Using the analogy of encryption, it is possible to decrypt the original template from the cancellable template by using the parameter. In other words, an attacker, with a cancellable template and a corresponding parameter, can obtain the original template using these transformations. Therefore, it is important to manage the parameter securely and separately from the cancellable template in order not to compromise the security of the encryptions and original templates simultaneously. Even if one of the two data, i.e., the parameter or the cancellable template, is compromised, it is possible to recover security by revoking and replacing both data, i.e., changing the parameter and replacing the cancellable template.

We may note, in passing, that there are many studies on one-way or non-invertible trans- formations for cancellable biometrics, such as [27, 33, 36, 21, 37, 39]. These studies aim at constructing transformations which make it sufficiently hard to recover the original template, even if both the cancellable template and the corresponding parameter are known. However, recent studies show vulnerabilities in the sense that it is easy to find either a close approximation of the original template or a pre-image of the cancellable template [20, 16, 17]. Note that the original template is not necessary, but one of the pre-images (or one similar to it) is sufficient for impersonation attack [17]. The difficulty of finding a biometric feature from a cancellable template that is close enough to “match” the original

template is called *authorised-leakage irreversibility* in [26]. As pointed out in [26], breaking authorised-leakage irreversibility is not difficult unless the FAR is extremely low. Otherwise, an attacker can perform an *offline FAR attack* as follows; for each sample from a sufficiently large biometric database of real or artificially generated features, the attacker transforms the sample, and compares it to the cancellable template. To prevent attacks to authorised-leakage irreversibility, including the offline FAR attack, again, data separation is recommended even for one-way or non-invertible transformation.

The rest of this chapter is organized as follows. Sec. 2 is an overview of the BTP and the cancellable biometrics scheme. In Sec. 3, several algorithms for cancellable biometrics as examples are reviewed. In Sec. 4, naive parameter management schemes for cancellable biometrics are introduced, and their limitations, mainly of usability of authentication systems, are discussed. In Sec. 5, another parameter management scheme with high usability and security is introduced, which is based on a server-side parameter management model and an authentication protocol using one-time parameters and one-time templates. In Sec. 6, the security of the introduced scheme is evaluated, and its usability is compared with other schemes. Finally, the chapter is summarized in Sec. 7.

2 Biometric Template Protection and Cancelable Biometrics

2.1 Architecture Overview

An overview of BTP architecture described in the ISO/IEC24745 is provided in Fig.1. During enrollment, the extracted biometric feature is encoded by a pseudonymous identifier encoder (PIE) to generate a pseudonymous identifier (PI) and auxiliary data (AD). The PI and AD pair is called a renewable biometric reference (RBR). During authentication, the newly extracted biometric feature is transformed to a pseudonymous identifier (PI*) by a pseudonymous identifier recorder (PIR). Following this, the pseudonymous identity comparator (PIC) compares PI and PI* and returns a similarity score.

In the context of cancellable biometrics, an AD is called a parameter and a PI is called a cancellable template. As discussed above, the cancellable template (PI) and the parameter (AD) should be stored and managed separately, in order to avoid being compromised simultaneously. In the ISO/IEC24745, eight system models (Models A to H) with different scenarios for the storage of PIs and ADs are listed [11]. However, data

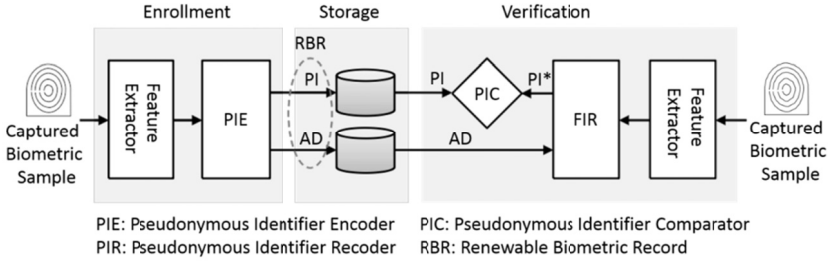

Figure 1: Architecture for biometric template protection [11]

separation is considered only in two models (Models G and H). Furthermore, in one of the two models (Model H), where the PI is stored in a client and the AD is stored in a token, the client reads the AD from the token at the authentication stage. Therefore, the PI and AD may leak immediately from a malicious or vulnerable client. In this sense, this model is not a secure data separation model for BTP.

A typical data separation model for cancellable biometrics and data flow is shown in Fig.2.

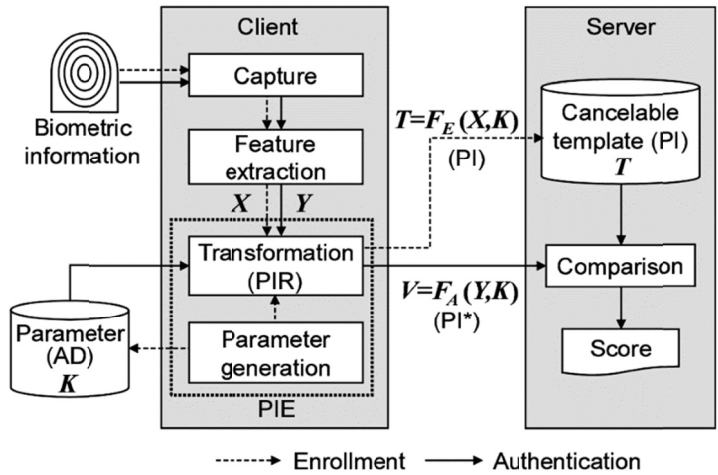

Figure 2: Typical system model of cancellable biometrics

Let X, Y denote biometric features for enrollment and authentication respectively, and K, T denote a parameter and a cancellable template respectively. A PIE can be constructed using a parameter generation function *Gen* and a feature transformation function, F_E. Typically, *Gen* generates a parameter K randomly using, for example, a pseudo random generator. Here, a PIR, of a feature transformation function, F_A, can be constructed. The transformation functions F_E and F_A, which can be the same (e.g., [35]) or different (e.g., [29]), are defined as follows:

$$F_E, F_A: \mathcal{X} \times \mathcal{K} \rightarrow \mathcal{T}, \tag{1}$$

where $\mathcal{X}$ is the biometric feature space, $\mathcal{K}$ is the parameter space, and $\mathcal{T}$ is the transformed feature space.

In the enrollment stage, a biometric feature data X is transformed to a cancellable template $T = F_E(X, K)$ based on a randomly generated parameter K, and stored in the server as a PI. K is stored in a client-side storage device such as a USB token or a smart card. In the authentication stage, a newly extracted feature Y is transformed to $V = F_A(Y, K)$ using the parameter K retrieved from the storage and sent to the server as a PI*. The server compares V and T and evaluates the similarity. The transformation functions F_E and F_A can be the same or different. Even if the cancellable template $T = F_E(X, K)$ or the parameter K leaks out, either can be revoked by generating a new parameter K' and replacing T with $T' = F_E(X, K')$.

2.2 Desirable Properties

Desirable properties or criteria for performance evaluation of BTP systems have been considered and discussed in literature, for example [13, 17, 6], and organized in [26] into three categories of performance: technical, protection, and operational.

The technical performance includes accuracy (or accuracy degradation), throughput, and so on. The operational performance includes modality independence, interoperability, and so on. Refer to [26] for the detail.

In this chapter, we focus on the protection performance or security, i.e., *irreversibility* and *unlinkability*. Irreversibility refers to the secrecy of the original biometric feature from the renewable biometric reference RBR=(PI,AD) or the PI alone or the AD alone. This property is subdivided into (i) full-leakage irreversibility (FLI), and (ii) authorized-leakage

irreversibility (ALI)[1]. The FLI refers to a difficulty to determine the exact original feature, whereas ALI refers to a difficulty to determine a feature similar to the original feature adequate to pass authentication. From a security point of view, the ALI is more important than the FLI. However, as mentioned above, if an attacker knows the RBR=(PI,AD), the ALI cannot be achieved in practice, due to the effect of the offline FAR attack, unless the FAR is extremely low. Therefore, here, we discuss the ALI from the PI alone or the AD alone.

Alternatively, *unlinkability* refers to the difficulty of cross-comparison of the RBRs or the PIs or the ADs, and determines if they are generated from the same biometric feature or not. If the operators of the systems collude with each other, they may be able to relate the user ID of each system by cross-comparing the DBs. Unlinkability is necessary so to prohibit successful cross-comparison, and to protect the privacy of users who have enrolled the RBRs to different systems. As is the case with irreversibility, attackers who know the RBR1 = (PI1, AD1) and RBR2 = (PI2 , AD2) can perform an offline FAR attack to break unlinkability: For each sample from a sufficiently large biometric database, the attacker tries to transform and match it against each RBRi (i = 1, 2). If the attacker finds a sample which matches both RBR1 and RBR2, he/she can guess that these are from the same biometric feature with high probability. Therefore, as well as irreversibility, we discuss unlinkability from the PI alone or the AD alone.

3 Examples of Cancelable Biometrics

3.1 Geometric Transformation

Ratha *et al.* proposed several feature transformation functions for minutiae matching-based cancellable fingerprint templates, i.e., *Cartesian, polar* and *functional transformations* [21].

We assume that fingerprint features X are represented as *minutiae*: a set of feature points $X = \{(x_i, y_i, \theta_i) | i = 1, \cdots, n\}$ where (x_i, y_i) and θ_i are the coordinates and the ridge direction of the i-th feature point extracted from a fingerprint image. The origin of the coordinate system is set based on the position of a singular point, such as the *core* of the fingerprint.

[1] Although another property: pseudo-authorized-leakage irreversibility (PLI) is defined in [26], we do not distinguish the PLI from the ALI to reduce argument.

3.1.1 Cartesian Transformation

The Cartesian transformation divides the feature space, i.e., the fingerprint image region, into $N = N_x \times N_y$ cells of fixed size, after which the cell positions are shuffled. Fig.3 illustrates an example of a Cartesian transformation where $N = 5 \times 5$. In this case, for example, the 3rd and 14th cells are transformed to the same 9th cell. The transformation is not necessarily a strict permutation, and allows overlapping; more than one cell can be mapped to the same position. All the minutiae within each cell are moved along with the cell position, retaining their relative positions. For each minutiae $(x,\ y,\ \theta)$ within a cell position $c_i \in \{1,2,\cdots,N\}$, the transformation function can be written as follows:

$$
\begin{aligned}
x' &= x + P_x(c_i') - P_x(c_i),\\
y' &= y + P_y(c_i') - P_y(c_i),\\
\theta' &= \theta,
\end{aligned}
$$

where $(P_x(c_i), P_y(c_i))$ are the coordinates of the center of the c_i-th cell, and $c_i'$is the position where the c_i-th cell is mapped.

The cell mapping can be written as

$$\boldsymbol{c'} = K\boldsymbol{c} \tag{2}$$

where $\boldsymbol{c} = (c_1 \dots, c_N)^T$,$\boldsymbol{c'} = (c_1' \dots, c_N')^T$ and K is a mapping matrix of size $N \times N$. Each row vector of K contains only one "1" and the other elements are all "0": For example, in the case of Fig.3, $\boldsymbol{c} = (1, 2, 3, \dots, 25)^T$ and $\boldsymbol{c'} = (12, 3, 9, \dots, 14)^T$. This means that the 1st cell is transformed to the 12th position, the 2nd cell is transformed to the 3rd position, and so on.

The transformation functions for enrollment F_E and for authentication F_A are the same, and the mapping matrix K plays the role of a parameter for the transformation $T = F_E(X, K), V = F_A(Y, K)(= F_E(Y, K))$.

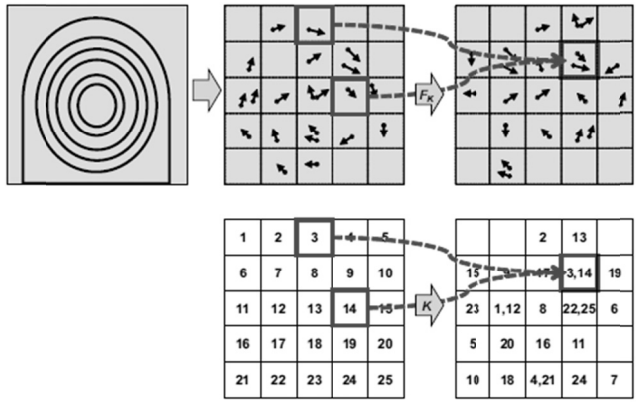

Figure 3: Cartesian transformation

The primary drawback of the Cartesian transformation is, as described in [21], the *boundary problem*: If an original minutiae point crosses a boundary of cells or sectors dividing the feature space due to minor deviation of image alignment or distortion of a fingerprint, then the transformed version of the minutiae point is located far from the appropriate position.

3.1.2 Functional Transformation

To avoid the boundary problem, the transformation function should be locally smooth. How- ever, if the minutiae positions after transformation are highly correlated, the transformation can be inverted easily. Thus, the transformation should not be globally smooth.

The third method, i.e., the functional transformations, is described as follows:

$$x' = x + f(x,y),$$
$$y' = y + g(x,y),$$
$$\theta' = \theta + h(x,y) \bmod 2\pi,$$

where f, g, and h are nonlinear perturbation functions. By designing f, g, and h appropriately, the above transformation becomes a "locally smooth but not globally smooth" function. See [21] for the details and examples of the function design. Figure 4 shows an example of the functional transformation.

Lee, et. al. [14] also proposed a locally smooth function for a cancellable fingerprint template which does not need alignment for the matching process.

3.2 Random Projection

Teoh *et al.* proposed *Biohashing* [35] for cancellable biometrics and applied this to fingerprints [35], the face [33], the palm [8], etc. Biohashing is based on a linear transformation of the feature vector

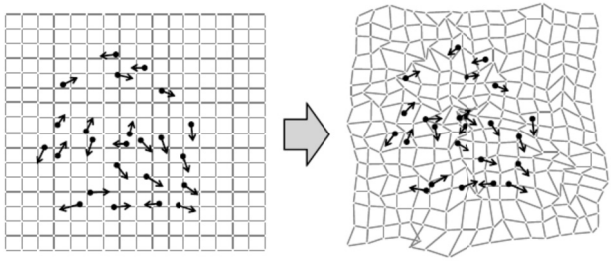

Figure 4: Functional transformation

from m dimensional space to $n(<m)$ dimensional subspace with a randomly selected basis, i.e., random projection.

Let us assume that a biometric feature $\boldsymbol{x} \in R^m$ is extracted as an m-dimensional vector. In

[33], for example, the Fisher Discriminant Analysis (FDA) is used for extracting feature vectors from face images. The Biohashing can be written as follows:

$$\boldsymbol{t} = \mathrm{Sig}(Kx - \tau \cdot 1) \tag{3}$$

where K is a user-specific $n \times m$ random matrix whose elements $K_{i,j}$ are independently and identically distributed (i.i.d.) according to a normal distribution $N(0, 1)$ and $\mathbf{1} = (1, 1, \ldots, 1)T$.

Sig: $\mathbb{R}^n \rightarrow \{0,1\}^n$ is defined as follows:

$$\mathrm{Sig}((y_1, \ldots, y_n)^T) = (t_1, \ldots, t_n)^T, t_i = \begin{cases} 0 & (y_i \leq 0) \\ 1 & (y_i > 0) \end{cases}$$

τ is a preset threshold and normally set to $\tau = 0$ [37]. Thus, hereafter, we assume $\tau = 0$. The Biohashing is used for the transformation function for enrollment F_E and for authentication F_A :

$$\boldsymbol{t} = F_E(\boldsymbol{x}, K) = \mathrm{Sig}(K\boldsymbol{x})$$
$$\boldsymbol{v} = F_A(\boldsymbol{y}, K) = \mathrm{Sig}(K\boldsymbol{y})$$

The matching decision is made based on the Hamming distance between the cancellable template $\boldsymbol{t}$ and the transformed feature $\boldsymbol{v}$. The Biohashing does not fully keep the distance structure between feature vectors, and the matching accuracy is inevitably degraded to some degree.

Chikkerur, et. al. [7] also proposed a transformation function for cancellable fingerprint tem- plates based on the random projection. Their method extracts a local image (called a *patch*) around each minutiae, and transforms it by a projection matrix which does not change the dot product measure of two patches.

3.3 Algebraic Transformation

Takahashi, et. al. proposed the correlation-invariant random filtering (CIRF) [29] which can be applied to construct cancellable biometrics for any kind of biometric authentication whose matching algorithm is based on the correlation-based template matching. In essence, the CIRF transforms a feature (typically an image) by convolution with a random image *K*, which plays a role as a parameter. To calculate the convolution, the CIRF utilizes the *number theoretic transform* (*NTT*) [19, 1], a kind of discrete Fourier transform (DFT) defined over a finite field Fq. Owing to some properties of the NTT, the CIRF fully keeps the matching accuracy as well as possessing information-theoretical security in the sense that the transformed feature does not leak any information about the original feature: The CIRF satisfies ALI. Hereafter, we review the CIRF.

Template matching is a well-known technique for image matching, which finds areas of an image, called a *search image*, that matches (i.e. is similar) to a certain small image, called a *template image* (see Fig.2). Template matching is used for various biometric verification systems such as the fingerprint [15], the face [4], and the iris [10].

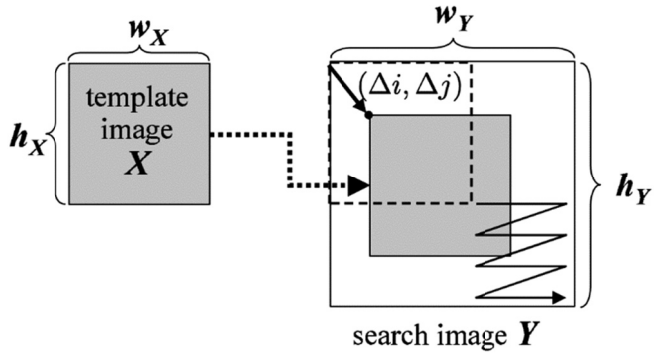

Figure 5: Template matching

Here, we assume that a biometric feature is represented as an image (i.e. a two-dimensional array of intensity values), each pixel value is an integer, and similarity is evaluated using cross- correlation.

Let $X[i,j](0 \le i < w_X, 0 \le j < h_X)$ be a template image of size $w_X \times h_X$, and $Y[i,j](0 \le i < w_Y, 0 \le j < h_Y)$ be a search image of size $w_Y \times h_Y$. We assume that $w_X \le w_Y, h_X \le h_Y$. The cross-correlation function X ⋆ Y is defined by

$$(X \star Y) = [\Delta\, i, \Delta\, j] = \sum_{i=0}^{w_X-1} \sum_{j=0}^{h_X-1} X[i,j]Y[i+\Delta\, i, j+\Delta\, j] \qquad (4)$$

The cross-correlation function $X \star Y$ can also be expressed in the following linear convolution formula:

$$(X \star Y)[\Delta i, \Delta j] = \left(X * \hat{Y}\right)[\Delta i, \Delta j] = \sum_{i=0}^{w_X-1} \sum_{j=0}^{h_X-1} X[i,j]\hat{Y}[w_Y - \Delta i - i - 1, h_Y - \Delta j - j - 1]$$

where $\hat{Y}$ denotes the flipped image of Y , i.e. $\hat{Y}[i,j] = Y[w_Y - i - 1, h_Y - j - 1]$, and $X * \hat{Y}$ denotes the linear convolution of X and $\hat{Y}$.

$(X \star Y)[\Delta i, \Delta j]$ indicates the cross-correlation value between the images X, Y when X is displaced by $(\Delta i, \Delta j)$ from Y. The displacement $(\Delta i, \Delta j)$ is allowed within the following region:

$$D = \{(\Delta i, \Delta j) | 0 \le \Delta i \le w_Y - w_X, 0 \le \Delta j \le h_Y - h_X\} \tag{5}$$

Here, we introduce the following transformation $\mathfrak{F}$: $\mathbb{F}_q^{mn} \to \mathbb{F}_q^{mn}$,

$$\mathfrak{F}(\tilde{X})[u,v] = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \omega_m^{ui} \omega_n^{uj} \tilde{X}[i,j] \bmod q \tag{6}$$

where q is a prime number and ωm, ωn are the elements of the Galois field $\mathbb{F}_q = \mathbb{Z}/q\mathbb{Z}$ whose orders are m, n respectively. It is assured that $\omega_m, \omega_n \in \mathbb{F}_q$ exist if $m, n | q-1$ i.e., m, n divide $q-1$. $\mathfrak{F}$ is a kind of DFT defined over $\mathbb{F}_q$, and called the number theoretic transform (NTT). Hereafter, let us assume all the numerical operations are performed over $\mathbb{F}_q$ and let us omit the notation "mod q", if not otherwise specified. It is well known that F has an inverse transformation $\mathfrak{F}^{-1}$ and has a *cyclic convolution property* (CCP) [1]:

$$\mathfrak{F}(\tilde{X} \circledast \tilde{Y}) = \mathfrak{F}(\tilde{X}) \circ \mathfrak{F}(\tilde{Y}) \tag{7}$$

where $\tilde{X} \circledast \tilde{Y}$ denotes the cyclic convolution:

$$\tilde{X} \circledast \tilde{Y}[\Delta i, \Delta j] = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \tilde{X}[i,j] \tilde{Y}[i', j'] \tag{8}$$
$$(i' = m - \Delta i - i - 1 \bmod m, j' = n - \Delta j - j - 1 \bmod n)$$

and $A \circ B$ denotes pixel-wise multiplication, i.e., $(A \circ B)[u,v] = A[u,v]B[u,v]$.

The CIRF makes use of the CCP of NTT to calculate the cross-correlation for template matching. Firstly, the size of the images $X, \hat{Y}$ is extended to $m \times n$, where m, and n are any integers satisfying $m, n \mid q-1$ and $w_Y \le m, h_Y \le n$. The extended area is padded with zeros. Let $\tilde{X}, \tilde{Y}$ be the extended images. Secondly, $\mathfrak{F}$ is applied to the extended images, and then, transformed by using an image of size $m \times n$ whose pixels are all non-zero random values in $\mathbb{F}_q$ (i.e., $K[u,v] \in \mathbb{F}_q^*$ where $\mathbb{F}_q^* = \mathbb{F}_q - \{0\}$) as follows:

$$T = F_E(X,K) = \mathfrak{F}(\tilde{X}) \circ K, \quad V = F_A(Y,K) = \mathfrak{F}(\tilde{Y}) \circ K^{-1} \tag{9}$$

where $K^{-1}[u,v] = K[u,v]^{-1}$. T plays a role as a cancellable template generated in the enrollment stage, and V as a transformed feature in the authentication stage. K is called a *random filter*, and plays the role of a parameter. We can calculate the cyclic convolution $\tilde{X} \circledast \tilde{Y}$ from T and V as follows:

$$\mathfrak{F}^{-1}(T \circ V) = \mathfrak{F}^{-1}\left(\mathfrak{F}(\tilde{X}) \circ \mathfrak{F}(\tilde{Y})\right) = \tilde{X} \circledast \tilde{Y}. \tag{10}$$

Since the extended areas of $\tilde{X}$ and $\tilde{Y}$ are padded with 0 and do not contribute to the calculation of the cyclic convolution, the linear convolution (5), and hence the cross-correlation $X \star Y$ (4), within the region D (5), can be calculated exactly. Therefore, the CIRF does not degrade the accuracy performance of the template matching.

Furthermore, as for the security, the following theorems hold.

Theorem 1 (Irreversibility)**.** *Let* $T = F_E(X,K)$. *If* $\tilde{X}$ *does not contain zero pixels, i.e.,* $\mathfrak{F}(\tilde{X})[u,v] \neq 0$ *for all* (u, v) *(*1),*

$$\Pr(X|T) = \Pr(X) \Leftrightarrow \mathrm{I}(X;T) = 0. \tag{11}$$

$\mathrm{I}(X; T)$ denotes the mutual information between X and T. Refer to [30] for the proof. This theorem indicates that the cancellable template T does not leak any information about the original feature $\tilde{X}$, i.e., the CIRF satisfies the ALI. The same property holds for $V = F_A(Y,K)$, i.e., $\mathrm{I}(Y; V) = 0$.

Theorem 2 (Unlinkability)**.** *Let* $T_1 = F_E(X,K_1)$ *and* $T_2 = F_E(X,K_2)$. *If the same condition (*1) as in the Theorem.1 holds,*

$$\Pr(T_1|T_2) = \Pr(T_1) \Leftrightarrow \mathrm{I}(T_1;T_2) = 0. \tag{12}$$

Refer to [30] for the proof. This theorem means that two cancellable templates T_1, T_2 generated from the same biometric feature are statistically independent, thereby they have no correlation.

The primary limitation of the CIRF is that the proof of irreversibility and the unlinkability require the condition (*1) in reference to the original feature image. In [31] this problem is solved by generalizing the CIRF based on a quotient polynomial ring.

4 Naive Parameter Management Schemes

In this section, we explain three naive parameter management schemes based on the following system models: (1) Store on Client model, (2)

Store on Token model and (3) Password-Based Parameter Generation model, and describe enrollment and authentication protocols for each model. Hereafter, we simply refer to the parameter management schemes based on each system model and set of protocols as SOC, SOT and PBPG schemes.

4.1 Store on Client

In the SOC scheme, the parameter is stored and managed in a client such as a PC, a mobile terminal or a sensor device. Enrollment and authentication protocols for the SOC model are as follows.

Enrollment protocol for the SOC model

1. A user inputs his/her ID and biometric information to an enrollment client.
2. A parameter K is chosen by the enrollment client and stored in the authentication client associated with the ID.
3. The enrollment client extracts a template X from the user's biometric information, transforms it to $T = F_E(X, K)$ and sends it to the authentication server.
4. The authentication server stores the cancellable template T associated with the ID.

Authentication protocol for the SOC model

1. A user inputs his/her ID and biometric information to an authentication client.
2. The authentication client extracts a biometric feature data Y, transforms it to $V = F_A(Y, K)$ using the parameter K associated with the ID, and sends it to the authentication server.
3. The authentication server matches the transformed feature V to the cancellable template T to decide acceptance or rejection.

Unlike the SOT scheme and the PBPG scheme described in the following subsections, the SOC scheme does not need a hardware token or a password. However, if the clients are shared by a large number of users, such as is the case with bank ATMs, POS and kiosk terminals, each client has to store and manage the parameters of all the potential users. In this case, if only one of the authentication clients is compromised, all the parameters in all the clients have to be revoked at once, which would require a large operational cost. It should be noted that the risk of compromise is proportional to the number of clients. For this reason,

authentication clients available to a user should be limited to only a few predetermined ones. This limitation may reduce the usability of the authentication system. The SOC scheme is discussed in, for example, [3].

4.2 Store on Token

In the SOT scheme, the parameter is stored in a hardware token such as a smart card or a USB token, and managed by each user. Enrollment and authentication protocols for the SOT model are as follows.

Enrollment protocol for the SOT model

1. A user inputs his/her ID and biometric information to an enrollment client.
2. A parameter K is chosen by the enrollment client and stored in a hardware token.
3. The enrollment client extracts a template X from user's biometric information, transforms it to $T = F_E(X, K)$, and sends it to the authentication server.
4. The authentication server stores the cancellable template T associated with the ID.

Authentication protocol for the SOT model

1. A user inputs his/her ID and biometric information to an authentication client.
2. The authentication client reads the parameter K from the token, extracts biometric feature data Y, transform it to $V = F_A(Y, K)$, and sends it to the authentication server.
3. The authentication server matches the transformed feature V to the cancellable template T so to decide acceptance or rejection.

The SOT scheme can be viewed as two-factor authentication using a hardware token and biometrics if it is sufficiently hard to impersonate a user without knowing both the biometric feature and the parameter. From another point of view, however, the SOT scheme reduces the usability of the authentication system because it requires a user to carry a hardware token which is easily misplaced. The SOT scheme is discussed in, for example, [34].

4.3 Password-Based Parameter Generation

The PBPG scheme is similar to well-known *password-based encryption* (PBE) [24]. In this scheme, the parameter is generated from a user's secret knowledge, such as a password. Enrollment and authentication protocols for the PBPG model are as follows.

Enrollment protocol for the PBPG model

1. A user inputs his/her ID, password and biometric information to an enrollment client.
2. The enrollment client generates a parameter K from the password using e.g., a secure hash function, extracts a template X from user's biometric information, transforms it to $T = F_E(X, K)$, and sends it to the authentication server.
3. The authentication server stores the cancellable template T associated with the ID.

Authentication protocol for the PBPG model

1. A user inputs his/her ID, password and biometric information to an authentication client.
2. The authentication client generates a parameter K from the password, extracts a biometric feature data Y, transforms it to $V = F_A(Y, K)$ using the parameter K associated with the ID, and sends it to the authentication server.
3. The authentication server matches the transformed feature V to the cancellable template T so to decide acceptance or rejection.

As with the SOT scheme, the PBPG scheme can also be viewed as two-factor authentication using passwords and biometrics if it is sufficiently hard to impersonate a user without knowing both the biometric feature and the parameter. Note, however, easy-to-remember passwords will not have enough complexity against dictionary attacks to recover the original feature from the transformed one. Sufficiently complex passwords are required to secure the template, which would reduce the usability of the authentication system.

5 Another Parameter Management Scheme for Cancelable Biometrics

In the previous section, we described the three naive schemes for parameter management of cancellable biometrics. However, they all have limitations in terms of usability: the SOC scheme limits a user to using individually predetermined authentication clients, the SOT scheme requires a user to carry a hardware token, and the PBPG scheme requires a user to remember a sufficiently complex password.

In this section, we will introduce another parameter management scheme that meets the following requirements.

(i) It should not require a user to carry a hardware token or to remember a password for authentication.
(ii) It should enable users to use any client connected to the system for authentication.
(iii) It should keep the parameters secure, irrespective of the number and vulnerabilities of the clients.

To this end, we consider another system model, i.e., the Store on Server (SOS), where a parameter management server is used in addition to the authentication server. As we show, however, a naive authentication protocol for this model does not satisfy the requirement (iii) and degrades the security of cancellable biometrics. To address this issue, a secure authentication protocol based on one-time parameters and one-time templates is introduced.

5.1 Store on Server

Fig. 6 shows an overview of the SOS model. The authentication system consists of enrollment clients, authentication clients, an authentication server, and a parameter management server. The parameter management server stores the parameters of all users, while the authentication server stores the cancellable templates, both associated with the user IDs.

We assume that the following requirements are fulfilled.

(A1) The authentication server and the parameter management server are administered separately by different administrators or organizations, and they do not collude with each other. This requirement is necessary because if the parameters and cancellable templates are compromised at once, the FAR attack can be performed.

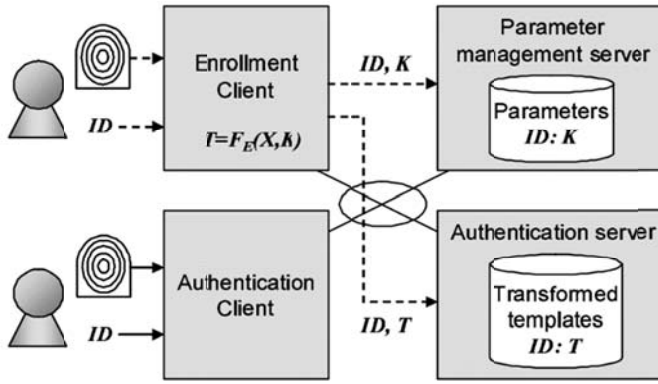

Figure 6: Store on Server

(A2) The communication channel between each pair of entities of the system (e.g., between an authentication client and the parameter management server, between the authentication server and the parameter management server and so on) is encrypted independently, e.g. by SSL. Thus, for example, the parameter management server cannot eavesdrop on the communication between an authentication client and the authentication server. This requirement is necessary to prevent recovery of the original biometric features or templates from the transmitted data over the channel.

(A3) The enrollment clients are securely managed and trustworthy.

(A4) The authentication clients are *tamper evident* [12] so that users or operators can easily find unauthorized alternations, e.g. by security seals, so to detect physical tampering and digital signatures to detect software tampering. Thus, we assume that the risk is small for biometric information to be compromised at an altered client used by a legitimate user during authentication. Note, however, an attacker may utilize an altered client to obtain some information from the servers by executing the authentication protocol.

The enrollment protocol for the SOS model is as follows:

Enrollment protocol for the SOS model

1. A user inputs his/her ID and biometric information to an enrollment client.
2. The enrollment client chooses a parameter K randomly and sends it to the parameter management server.