2014

Efficient Anonymous Biometric Matching in Privacy-Aware Environments

Ying Luo
University of Kentucky, ying.luo2@gmail.com

STUDENT AGREEMENT:

I represent that my thesis or dissertation and abstract are my original work. Proper attribution has been given to all outside sources. I understand that I am solely responsible for obtaining any needed copyright permissions. I have obtained needed written permission statement(s) from the owner(s) of each third-party copyrighted matter to be included in my work, allowing electronic distribution (if such use is not permitted by the fair use doctrine) which will be submitted to UKnowledge as Additional File.

I hereby grant to The University of Kentucky and its agents the irrevocable, non-exclusive, and royalty-free license to archive and make accessible my work in whole or in part in all forms of media, now or hereafter known. I agree that the document mentioned above may be made available immediately for worldwide access unless an embargo applies.

I retain all other ownership rights to the copyright of my work. I also retain the right to use in future works (such as articles or books) all or part of my work. I understand that I am free to register the copyright to my work.

REVIEW, APPROVAL AND ACCEPTANCE

The document mentioned above has been reviewed and accepted by the student's advisor, on behalf of the advisory committee, and by the Director of Graduate Studies (DGS), on behalf of the program; we verify that this is the final, approved version of the student's thesis including all changes required by the advisory committee. The undersigned agree to abide by the statements above.

Ying Luo, Student

Dr. Sen-Ching S. Cheung, Major Professor

Dr. Caicheng Lu, Director of Graduate Studies

EFFICIENT ANONYMOUS BIOMETRIC MATCHING IN PRIVACY-AWARE ENVIRONMENTS

DISSERTATION

A dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
in the College of Engineering
at the University of Kentucky

By

Ying Luo

Lexington, Kentucky

Director: Dr. Sen-Ching S. Cheung, Professor of Electrical and Computer

Engineering

Lexington, Kentucky

2014

ABSTRACT OF DISSERTATION

EFFICIENT ANONYMOUS BIOMETRIC MATCHING IN PRIVACY-AWARE ENVIRONMENTS

Video surveillance is an important tool used in security and environmental monitoring, however, the widespread deployment of surveillance cameras has raised serious privacy concerns. Many privacy-enhancing schemes have been recently proposed to automatically redact images of selected individuals in the surveillance video for protection. To identify these individuals for protection, the most reliable approach is to use biometric signals as they are immutable and highly discriminative. If misused, these characteristics of biometrics can seriously defeat the goal of privacy protection. In this dissertation, an Anonymous Biometric Access Control (ABAC) procedure is proposed based on biometric signals for privacy-aware video surveillance. The ABAC procedure uses Secure Multi-party Computational (SMC) based protocols to verify membership of an incoming individual without knowing his/her true identity. To make SMC-based protocols scalable to large biometric databases, I introduce the k-Anonymous Quantization (kAQ) framework to provide an effective and secure tradeoff of privacy and complexity. kAQ limits systems knowledge of the incoming individual to k maximally dissimilar candidates in the database, where k is a design parameter that controls the amount of complexity-privacy tradeoff. The relationship between biometric similarity and privacy is experimentally validated using a twin iris database. The effectiveness of the entire system is demonstrated based on a public iris biometric database.

To provide the protected subjects with full access to their privacy information in video surveillance system, I develop a novel privacy information management system that allows subjects to access their information via the same biometric signals used for ABAC. The system is composed of two encrypted-domain protocols: the privacy information encryption protocol encrypts the original video records using the iris pattern acquired during ABAC procedure; the privacy information retrieval protocol allows the video records to be anonymously retrieved through a GC-based iris pattern matching process. Experimental results on a public iris biometric database demonstrate the validity of my framework.

KEYWORDS: Biometric Matching, Privacy, Authentication, Anonymity, Access Control

Author's signature: Ying Luo

Date: July 3, 2014

EFFICIENT ANONYMOUS BIOMETRIC MATCHING IN PRIVACY-AWARE ENVIRONMENTS

By

Ying Luo

Director of Dissertation: Sen-Ching S. Cheung

Director of Graduate Studies: Caicheng Lu

Date: July 3, 2014

For my family

ACKNOWLEDGMENTS

I would never have been able to finish my dissertation without the guidance of my committee members, help from friends, and support from my family.

I am heartily thankful to my supervisor, Dr. Sen-Ching Samson Cheung, whose encouragement, guidance and support from the initial to the final level enabled me to develop an understanding of the subject. I also want to give many thanks to my committee members and outside examiner: Dr. Kevin Donohue , Dr. Laurence Hassebrook, Dr. Jun Zhang and Dr. William Murphy, for the time they spent on my dissertation review. I appreciate all of the suggestions and comments.

I want to thank Dr. Mauro Barni and his Visual Information Processing and Protection (VIPP) group at the Department of Information Engineering of the University of Siena in Italy. He not only introduced garble circuits to improve the efficiency of my proposed system, but also infected me with his passion and altitude towards research.

In addition, special thanks for the financial support from the National Science Foundation under Grant No. 1018241 & No. 1237134, and Dr. Bruce Walcott's project sponsored by Lockheed Martin.

Last but not least, my deepest gratitude goes to the people who have had such a significant impact on my life. My parents and elder brother encouraged me to enter a Ph.D. program in the United States. They have always given me unconditional support and love.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, surveillance cameras have been widely used for preventing theft, collecting population data, and combating terrorism. Advances in pattern recognition algorithms such as searchable surveillance and automatic event/human recognition have turned the once labor-intensive processes into powerful automated systems that can quickly and accurately identify visual objects and events. From the public outcry on the use of face recognition in public events [1] to the report by the American Civil Liberties Union (ACLU) on the surveillance systems' assault on public's privacy [2], it is unsurprising that the general public is increasingly wary about the possibility of privacy invasion with video surveillance systems.

To mitigate the public's concern and to facilitate continued development of surveillance technologies, it is imperative to make privacy protection a priority in current and future video surveillance systems. Most of the research effort for privacy protection in surveillance systems has been devoted to visually obfuscate the images of individuals for protection. They range from the use of black boxes or large pixels in [3, 4], scrambling in [5] to complete object removal in [6, 7]. Some examples are shown in Figure 1.1.

Most of the obfuscation schemes apply a blanket protection to every individual

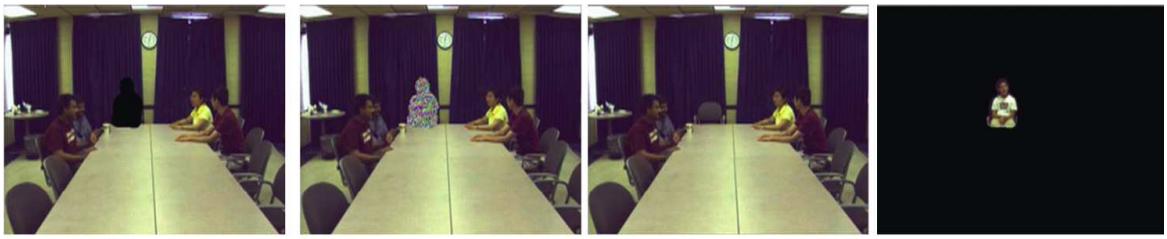


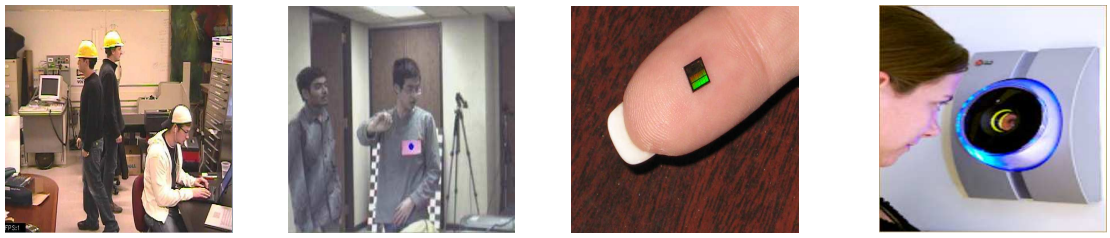
Figure 1.1: Different visual obfuscation techniques: (a) black silhouette; (b) scrambling; (c) complete removal; (d) original information. Graphics adapted from original in [8].

in the scene. For such a strategy to work, the obfuscated video must reveal some attributes such as a body with a blurred face or a moving blob otherwise the video would be useless for surveillance. Total anonymity cannot be achieved due to the release of partial information of visual objects. This type of privacy protection is not defensible in any rigorous sense of security. It is only adequate for public places such as airports or banks where there is no reasonable expectation of privacy.

There are many situations in shared environment where a group of "trusted" individuals have certain expectation of privacy. For example, privacy of students in school is protected under the Family Educational Rights and Privacy Act (FERPA) of 1974 in the United States. Similarly, patients in a clinic or hospital enjoy the same protection as guaranteed by the Health Insurance Portability and Accountability Act (HIPPA) of 1996. Even in commercial entities such as department stores or apartment buildings, patrons will likely to stay away from vendors who abuse surveillance cameras in monitoring every move the patron makes. In order to provide privacy protection for these applications, it is imperative to have a reliable mechanism to identify whether an individual belongs to this group. While members of this group

will enjoy total anonymity in the surveillance system, other transient visitors must be monitored at all time.

To identify the subjects for privacy protection, as shown in Figure 1.2, there are two general approaches: one is to use special markers such as yellow hard-hats [9], visual tags [7], or RFID [6]; the other relies on biometric signals such as faces [10], skin tones [3], or irises [11]. Unfortunately, both approaches have their shortcomings.



(a) Yellow hat [9] (b) Visual tag [7] (c) RFID [12] (d) Iris scan [13]

Figure 1.2: Existing subject identification approaches

The first approach requires the protected subjects to carry a special marker. If the marker is accidentally dropped, the subjects will lose privacy protection from the system. If the marker is maliciously embezzled by unauthorized individuals, the system will protect the potential intruders and the security of the environment will be severely compromised.

On the other hand, the second approach uses biometric signals which excel in authenticating the subject's identity as biometric signals are based on "who you are" rather than "what you have". While the use of biometrics enhances system security and alleviates users from carrying identity cards or remembering passwords, it creates a conundrum for privacy advocates as the knowledge of the identity makes it much

harder to keep users anonymous. A curious system operator or a parasitic hacker can infer the identity of a user based on his/her biometric probe. Furthermore, as biometrics is immutable from systems to systems, it can be used by attackers to cross-correlate disparate databases and cause damages far beyond the coverage of any protection schemes for individual database systems. The use of biometric signals provides *a direct link between the imagery to the true identity of an individual without even using any sophisticated pattern recognition algorithms*. If the security of the system is compromised, this extra information may create greater privacy concerns that it purports to protect. It is thus important to sever this link between the biometric signal and the imagery. To take advantage of the superior performance of biometric technologies, it is imperative to strengthen the security of the surveillance systems to protect the anonymity of the subjects.

1.1 Problem Statement

In this dissertation, I study two problems related to biometric matching and privacy-aware video surveillance. The first problem is how one can reliably identify trusted individuals for privacy protection without revealing sensitive biometric data. I call this problem the Anonymous Biometric Access Control (ABAC) problem. The second problem focuses on various technical issues to incorporate ABAC in the next generation privacy aware video surveillance system that allows each trusted individual to control how his/her imageries can be accessed. We call this problem the Privacy Information Management (PIM) problem.

The are three main challenges in developing solutions for the ABAC and PIM

problems. To simplify the description, I adopt the tradition from cryptography and refer the biometric reader as Alice and the server as Bob. First, to cope with the variability of the input probe, any biometric access system needs to perform a signal matching process between the probe and all the records in the database. The challenge here lies in making the process private so that Bob can confirm the membership status of Alice without knowing any additional information about Alice's probe. It is imperative to prevent Bob from extracting any knowledge about Alice's probe and its similarity distances with any records in Bob's database. On the other hand, Bob must be able to compare the distances to a similarity threshold and prevent Alice from cheating her membership status. This problem is an instance of secure multiparty computation, a subfield of cryptography in which multiple parties use the private data to achieve a common computation goal [14].

Second, we consider the complexity challenge posed by scaling the biometric matching process to large databases through secure collaboration between Alice and Bob, normally in encrypted form. The high complexity of cryptographic primitives is often cited as the major obstacle of their widespread deployment in realistic systems [15, 16, 17, 18, 19, 20]. This is particularly important for biometric applications that require matching a large number of high-dimensional feature vectors in real time. My approach in addressing this problem is to exploit the specific nature of the biometric process and develop a rigorous tradeoff between computational complexity and privacy.

Third, the ultimate goal of a privacy-aware surveillance system is to treat the privacy visual information of an individual in the same manner as any other privacy

information such as personal financial or medical information – each access of the information must require a full consent from the corresponding user. This posts a technical challenge because the surveillance system cannot associate the imagery with the unknown identity of the individual as protected by the ABAC process. On the other hand, the fact that we can anonymously match biometric signals implies that the biometric signal itself can be used as an encryption key for the private data. This is the approach that I have adopted in tackling the PIM problem.

1.2 Contributions of Dissertation

The research work presented in this dissertation addresses the challenges on user anonymity with biometric signals and the high complexity associated with state-of-the-art cryptographic solutions. Specifically,

1. To address the first challenge in making the biometric matching secure between Bob and Alice, I treat the matching process as an instance of Secure Multi-party Computational (SMC) protocol, which guarantees the privacy of both the biometric gallery and the probe. Though SMC has been used in solving relatively straightforward comparison problems such as Secure Millionaire Problem [21] electronic voting [22], online auction [23], keyword search [24], and anonymous routing [25], I am the first to apply SMC to biometric matching [26]. In this work, I proposed a Homomorphic Encryption (HE)-based protocol to the well-known approach by Daugman in matching iris-codes [27]. The initial work on using HE was computationally intensive. One reason is that HE

is cumbersome in handling binary operations needed for the hamming distance calculations in iris-code matching. Collaborating with Mauro Barni, I have provided an alternative implementation using Garbled Circuit (GC) [28]. This work also exploits key characteristics of iris data and results in one of the fastest anonymous iris matching at the time. Both HE and GC are computationally secure schemes and their security hinges on the use of large integer field which is another significant source of complexity. Recently, I have explored the use of information-theoretic security protocols based on Shamir's Secret Sharing to further reduce the complexity in making basic signal processing operations secure [29].

2. To address the second complexity challenge posed by scalability of anonymous biometric matching, I collaborate with Shuiming Ye to propose a novel framework called k-ABAC to provide a controllable trade-off between privacy and complexity [30]. Despite the reduction in computational and communication complexity, computations in SMC remain highly complex and the current state-of-the-art simply cannot scale to large databases that contain tens of thousands biometric signals [31, 32, 33]. My k-ABAC system provides further complexity reduction in order to scale the operations up to large databases. This is similar to the well-known k-anonymity model [34] in that k is a controllable parameter of anonymity. However, the two approaches are fundamentally different – the k-anonymity model is a data disclosure protocol where Bob anonymizes the database for public release by grouping all the data into k-member clusters. In

a k-ABAC system, the goal is to *prevent Bob from obtaining information about the similarity relationship between his data and the query probe from Alice.* In order to minimize the knowledge revealed by any k-member cluster, k-ABAC uses novel grouping scheme called k-Anonymous Quantization (kAQ) that optimizes the *dissimilarity* among members in the same group. kAQ forbids similar patterns to be in the same group which might be a result of multiple registrations of the same person or from family members with similar biometric features. The kAQ process is carried out mostly in plaintext and is computationally efficient. Using kAQ as a pre-processing step, the subsequent encrypted-domain matching can be efficiently realized within the real-time constraint.

3. To address the third challenge in associating privacy video with the unknown identity of the "trusted" individual in video surveillance system, I propose a novel Privacy Information Management (PIM) system that uses biometric signals for encrypting and retrieving the privacy video [35]. There have been many recent works in enhancing privacy protection in surveillance systems [6, 3, 4, 36, 37, 10, 38]. Many of them share the common theme of identifying sensitive information and applying image processing schemes for obfuscating that sensitive information. But the security flaw overlooked in most of these systems is that they fail to consider the security impact of modifying the surveillance videos. While sophisticated privacy policy has been studied in the literature [39], the privacy visual information of an individual should be ideally treated in the same manner as any other personal information such as passport

or credit card numbers. That is, every access of such information must require a full consent from the corresponding individual. To satisfy this goal, PIM provides selective and anonymous access to the preserved privacy information in video surveillance systems using iris biometrics. In PIM, the iris pattern will be combined with a user specified passcode to encrypt and retrieve the privacy video.

1.3 Overview of My Solution

In this dissertation, a Identity and Information Management system is built under a privacy-protected video surveillance camera network. The two security goals of the system are to *protect private information at each component* as well as *to offer visual privacy protection and original video access to only those individuals authenticated by iris matching.* The collaboration assumes a *semi-honest security model* in which every component faithfully follows the protocol but attempts to infer as much information as possible about others based on the information exchanged.

The Identity and Information Management system consists of two parts: Anonymous Biometric Access Contril (ABAC) and Privacy Information Management (PIM). Their execution is distributed throughout every hardware component of the system. The structure of the ABAC system is shown in Figure 1.3. The two main processing units in ABAC are the server and the biometric reader. The server has a biometric database of M biometric signals $DB = \{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$, where $\mathbf{x}_i = (x_1^i, \ldots, x_n^i)^T$ is the biometric signal of member i. The biometric reader is used to capture biometric probes for carrying out the matching process. The reader also has a keypad for the

user to enter a passcode which is used in combination with the biometric probe to encrypt the privacy video associate with this individual.

There are two main processing components in ABAC: the preprocessing step and the matching step. While the matching step is executed for every probe captured by Alice, the preprocessing step is executed only once by Bob to compute a *publicly-available* quantization table based on a process called k-Anonymous Quantization (kAQ). The purpose of the public table is that, based on a joint secure-index selection of the table entry between Alice and Bob, Bob can significantly reduce the scope of the similarity search from the entire database DB to approximately k candidates. The k-Anonymous Quantization guarantees that (1) if there is an entry in Bob's database that matches Alice's probe, this entry must be among these candidates, (2) all the candidates are maximally dissimilar so as to provide the least amount information about Alice's probe, and (3) the public table discloses no information about Bob's database. The details of the k-Anonymous Quantization and the secure-index selection will be discussed in Section 5.1.

After computing the proper quantization cell index from the public table, Bob identifies all the candidates and then engages with Alice in a joint secret matching process to determine if Alice's probe resembles any one of the candidates. This process is conducted under the ABAC framework described in Section 4.2. We assume that there is an open network between Bob and Alice that will guarantee message integrity. Since only encrypted content are exchanged, there is no need for any protection against eavesdroppers. For each session, Alice will be responsible for generating the private and public keys for the encryption and sharing the public key with Bob.

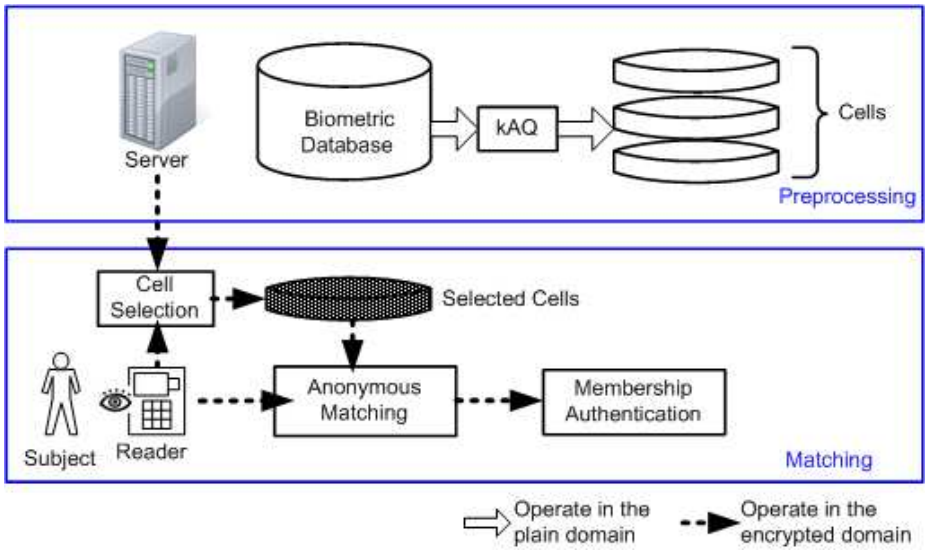
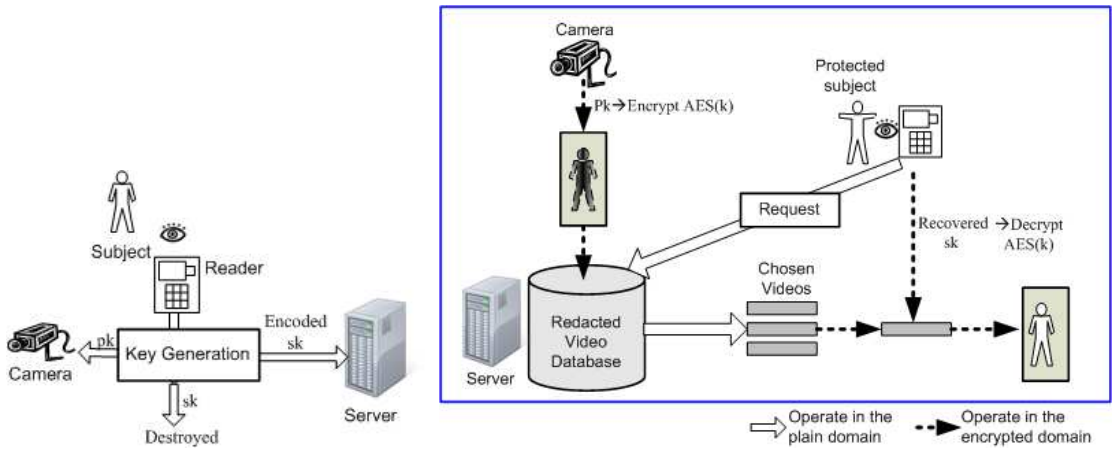


Figure 1.3: Anonymous Biometric Access Control

In other words, a different set of keys will be used for each different user. Furthermore this protocol demands comparable computational capabilities from both parties. Thus it is imperative to use the preprocessing step to reduce the computational complexity of this matching step. As the secret matching utilizes all the fundamental processing blocks of ABAC to implement anonymous subject identification system, we will first explain these building blocks in the following section.

Once a match is ascertained by the server, all the cameras in the camera network will be alerted to protect the imagery of the incoming individual and preserve the original video for later retrieval. The preservation and retrieval of privacy imagery are governed by the PIM component. First, the reader at the entrance needs to preserve the biometric signal in an anonymous fashion so that it can be used to encrypt the raw surveillance video associated with this individual. To accomplish this goal, a new pair of private-public keys are generated by the reader for each entry as shown

in Figure 1.4a. The public key is distributed to the camera network while the private key is encrypted using both the biometric signal and a personal passcode known only by the user. The encrypted private key is stored alongside with the encrypted video at the server. Details of this Privacy Information Encryption Protocol are given in Section 6.3.1.



(a) Key Generation and Distribution (b) Private Information Retrieval

Figure 1.4: Privacy Information Management

Second, the camera network is responsible for preserving the visual imageries of the protected individual throughout the entire surveillance area. As shown in Figure 1.4b, the preserved imageries are AES-encrypted using a random key which in turn is encrypted by the public key from the reader. The image processing component of the network is described in the next section. Third, as shown in the rest of Figure 1.4b, when there is a request to retrieve the original imageries, the requester must demonstrate his/her identity by presenting the time-and-day of entry, the correct biometric signal, and the passcode at an authenticated reader. A GC-based matching protocol is then executed between the reader and the server to recover the private

key which is needed to unlock the private video at that time period. Details of this Privacy Information Retrieval protocol can be found in Section 6.3.2.

1.4 Dissertation Organization

This dissertation is organized as follows: After introducing the basic concept of Anonymous Biometric Access Control (ABAC), and key cryptographic primitives used in my implementations in Chapter 2, I review previous techniques for preserving the privacy of biometric data while maintaining their usability in various scenarios in Chapter 3. In Chapter 4, an implementation of an ABAC system on iris biometrics under semi-honest security model is provided based on Homomorphic Encryption (HE) and Garbled Circuits (GC), and the possibility of using Information Theoretic SMC (IT-SMC) protocols, mainly with Shamir's Secret Sharing (SSS), is explored. In Chapter 5, inspired by the k-anonymity model, a simple approach is proposed to tradeoff complexity with privacy by quickly narrowing Alice's query into a small group of k candidates and then performing the full cryptographic search only on this small group. An application based on anonymous biometric matching in video surveillance network is implemented in Chapter 6. Finally, I conclude the dissertation and discuss future work in Chapter 7.

Chapter 2

Background

In this chapter, I first introduce the basic concept of Anonymous Biometric Access Control (ABAC), and then I provide an overview of the key cryptographic primitives and the optimization strategies used in our implementations of the anonymous subject identification system (Section 4.2) and privacy information management system (Section 6.3).

ABAC is implemented based on Secure Multi-party Computation (SMC) protocols, which are worked as follows: there are n parties $P_1, P_2, \ldots, P_n$ on a network, each party P_i has a private input x_i for $i = 1, 2, \ldots, n$. All parties want to compute a join function $f(x_1, x_2, \ldots, x_n)$ where P_i will not learn anything about other parties input beyond what can be inferred from her own private input and the final result of $f(x_1, x_2, \ldots, x_n)$ [14]. The two popular models in SMC are computationally-secure SMC (CS-SMC) and Information Theoretic SMC (IT-SMC) which will be covered in Section 2.2 and 2.3.

2.1 Anonymous Biometric Access Control (ABAC)

The Anonymous Biometric Access Control (ABAC) protocol is a SMC based protocol that supports anonymous matching of biometric signals between the biometric reader

at the entrance and the iris-database server. Following the tradition in describing any SMC protocols in Chapter 1, we refer the biometric reader as Alice and the server as Bob. Functionally, ABAC has the following guarantees:

1. The protocol returns a decision bit to Bob on whether the biometric signal of the incoming subject matches any entries in Bob's database;

2. No identity information of the subject is provided to Bob;

3. No database information is provided to Alice;

4. The communication between Bob and Alice is conducted over an open network.

The first and second guarantees define the anonymous subject identification process – Bob can reliably authenticate the privacy protection status of an incoming individual using biometric signals without knowing the actual identity. As the reader is installed outside the surveillance area, it is prone to outsider's attacks and thus should not possess any sensitive biometric signals as indicated in the third guarantee. To allow many readers to be used at all the entrances, the fourth guarantee implies that sensitive information is encrypted and can be transmitted via an open network without worrying about eavesdropper.

2.1.1 Biometric Signal Space and Distance Function

I model any biometric signal $\mathbf{x} = (x_1, \ldots, x_n)^T$ as a n-dimensional vector from a feature space F^n where F is a finite field. We also assume the existence of a commutative distance function $d : F^n \times F^n \to \Re^+ \cup \{0\}$ that measures the dissimilarity between

two biometric signals. In order for the distance to be computable using the operators in the field, we assume that F to be a subfield of $\Re$ so that the components of the constituent vectors will be treated as real numbers in the distance computation. The most commonly used distance is the Euclidean distance:

$$d(\mathbf{x},\mathbf{y})^2 :=\| \mathbf{x}-\mathbf{y} \|_2^2= \sum_{i=1}^{n}(x_i-y_i)^2 \tag{2.1}$$

For the iris patterns used in our experiments, F is the binary field $Z_2 = \{0,1\}$ and $d(\cdot,\cdot)$ is a modified hamming distance defined below [27]:

$$d_H(\mathbf{x},\mathbf{y})^2 := \frac{\|(\mathbf{x}\otimes\mathbf{y})\cap mask_{\mathbf{x}}\cap mask_{\mathbf{y}}\|_2^2}{\|mask_{\mathbf{x}}\cap mask_{\mathbf{y}}\|_2^2} \tag{2.2}$$

where $\otimes$ denotes the XOR operation and $\cap$ denote the bitwise AND. $mask_{\mathbf{x}}$ and $mask_{\mathbf{y}}$ are the corresponding mask binary vectors that mask the unusable portion of the irises due to occlusion by eyelids and eyelash, specular reflections, boundary artifacts of lenses, or poor signal-to-noise ratio.

The special distance function and the high dimension of many feature spaces make them less amenable to statistical analysis. There exist mapping functions that can project the feature space F^n into a lower dimensional space $\Re^m$ such that the original distance can be approximated by the distance, usually Euclidean, in $\Re^m$. The most well-known technique is Principal Component Analysis (PCA) which is optimal if the original distance is Euclidean [40]. For general distances, mapping functions can be derived by two different approaches – the first approach is Multi-dimensional Scaling (MDS) in which an optimal mapping is derived based on minimizing the differences between the two distances over a finite dataset [41]. The second approach is based

on distance relationship with random sets of points and include techniques such as Fastmap [42], Lipshcitz Embedding [43] and Local Sensitivity Hashing [44]. In our system, we use both PCA and Fastmap for their low computational complexity and good performance. Here we provide a brief review of the Fastmap procedure and will discuss its secure implementation in Section 5.1.4. Fastmap is an iterative procedure in which each step selects two random pivot objects $\mathbf{x}_A$ and $\mathbf{x}_B$ and computes the projection x' for any data point $\mathbf{x}$ as follows:

$$x' := \frac{d(\mathbf{x}, \mathbf{x}_A)^2 + d(\mathbf{x}_A, \mathbf{x}_B)^2 - d(\mathbf{x}, \mathbf{x}_B)^2}{2d(\mathbf{x}_A, \mathbf{x}_B)} \tag{2.3}$$

The projection in (2.3) requires only distance relationships. A new distance is then computed by taking into account the existing projection:

$$d'(\mathbf{x}, \mathbf{y})^2 := d(\mathbf{x}, \mathbf{y})^2 - (x' - y')^2 \tag{2.4}$$

where x' and y' are the projections of $\mathbf{x}$ and $\mathbf{y}$ respectively. The same procedure can now be repeated using the new distance $d'(\cdot, \cdot)$. It has been demonstrated in [42] that using pivot objects that are far apart, the Euclidean distance in the projected space produces a reasonable approximation of the original distance of many different feature spaces.

2.1.2 User Anonymity & Biometric Access Control

Using a dissimilarity metric, we can now define the function of a biometric access control system. It is a computational process that involves two parties: a biometric server (Bob) and a biometric reader (Alice). Bob is assumed to have a database of M biometric signals $DB = \{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$, where $\mathbf{x}_i = (x_1^i, \ldots, x_n^i)^T$ is the biometric

signal of member i. The reader or Alice is present outside each entrance of the surveillance area. She captures the iris pattern $\mathbf{q}$ of every individual entering the area. Once $\mathbf{q}$ is captured, Alice engages Bob in a specially designed Secure Multi-party Computational (SMC) protocol to determine if the incoming subject is an authorized individual. If so, Bob will activate the privacy protection mechanism and obfuscate the appearance of this subject after he/she enters the area. Armed with these notations, I first provide a functional definition of biometric access control.

Definition 1 *A* Biometric Access Control *(BAC) system is a computational protocol between two parties, Bob with a biometric database DB and Alice with a probe $\mathbf{q}$, such that at the end of the protocol, Alice and Bob can jointly compute the following value:*

$$y_{BAC} := \begin{cases} 1 & \text{if } d(\mathbf{q}, \mathbf{x}_i)^2 < \epsilon \text{ for some } \mathbf{x}_i \in DB \\ 0 & \text{otherwise.} \end{cases} \tag{2.5}$$

The use of distance square is to provide a consistent dimensionality for ϵ used in our implementation. Adding user anonymity to a BAC system results in the following definition:

Definition 2 *An* Anonymous BAC *(ABAC) system is a BAC system on DB and $\mathbf{q}$ with the following properties at the end of the protocol:*

1. *Except for the value y_{BAC}, Bob has negligible knowledge about $\mathbf{q}$, $d(\mathbf{q}, \mathbf{x})$, and the comparison results between $d(\mathbf{q}, \mathbf{x})^2$ and ϵ for all $\mathbf{x} \in DB$.*

2. *Except for the value y_{BAC}, Alice has negligible knowledge about ϵ, $\mathbf{x}$, $d(\mathbf{q}, \mathbf{x})$, and the comparison results between $d(\mathbf{q}, \mathbf{x})^2$ and ϵ for all $\mathbf{x} \in DB$.*

Like any other computationally secure protocols, "negligible knowledge" used in the above definition should be interpreted as, given the available information to a party, the distribution of all possible values of the private input from the other party is computationally indistinguishable from the uniformly random distribution [45]. The first property in Definition 2 defines the concept of user anonymity, i.e. Bob knows nothing about Alice except whether her probe matches one or more biometric signals in DB. As it has been demonstrated that even the distance values $d(\mathbf{q}, \mathbf{x}_i)$ are sufficient for an attacker to recreate DB [46], the second property is designed to disclose the least amount of information to Alice.

2.1.3 Security Model on Adversarial Behaviors

It is impossible to design a secure system without considering the possible adversarial behaviors from both parties. Adversarial behaviors are broadly classified into two types: semi-honest and malicious. A dishonest party is called semi-honest if he follows the protocol faithfully but attempts to find out about others' private data through the communication. A malicious party, on the other hand, will change private inputs or even disrupt the protocol by premature termination. Making the proposed system robust against a wide range of malicious behaviors is beyond the scope of this paper. Here, we assume Bob to be semi-honest but allow certain malicious behaviors from Alice – we assume that Alice will engage in malicious behaviors only if those behaviors can increase her chance of gaining access, that is turning y_{BAC} into 1, from using a purely random probe. This is a restricted model because, for example, Alice will not prematurely terminate before Bob reaches the final step in computing y_{BAC}.

Also, Alice will not randomly modify any private input unless such modification will increase her chance of success.

2.2 Computationally-Secure SMC (CS-SMC)

The secure computation of a joint function under Computationally-Secure SMC (CS-SMC) model, is also called secure two-party computation or Secure Function Evaluation (SFE), which only involves two partes. The secrets under this model are protected by encoding them based on a complicated mathematical function. It is impossible to compute the inverse of the mathematical function in polynomial time without any additional primitive [14].

One approach to protect the privacy of both the biometric server and the probe owner in our proposed system is to treat the matching process as an instance of SFE which guarantees the privacy of both the biometric gallery and the probe. The two prevailing approaches of implementing SFE are to use Homomorphic Encryption (HE) [47] and Garbled Circuits (GC) [48]. HE is an asymmetric public-key cipher that allows certain arithmetic operations such as addition to be directly performed on the encrypted data. GC provides a generic implementation of any binary function by having one party prepared an encrypted boolean circuit, and another party obliviously evaluated the circuit without access to intermediate values. Moreover, Oblivious Transfer (OT) as an important step for input exchanging in GC, is also introduced.

2.2.1 Homomorphic Encryption (HE)

An encryption system $Enc(x)$ is homomorphic with respect to an operation $f_1(\cdot,\cdot)$ in the plaintext domain if there exists another operator $f_2(\cdot,\cdot)$ in the ciphertext domain such that:

$$Enc(f_1(x,y)) = f_2(Enc(x), Enc(y)). \tag{2.6}$$

In our system, we choose the Paillier encryption system as it is homomorphic over a large additive plaitext group and thus providing a wide dynamic range for computation. Given a plaintext number $x \in Z_N$, the Paillier encryption process is given as follows:

$$Enc_{pk}(x) = \left[(1+N)^x \cdot r^N \bmod N^2\right] \tag{2.7}$$

where N is a product of two equal-length secret primes and r is a random number in Z_N to ensure semantic security. The public key pk consists of only N. The decryption function $Dec_{sk}(c)$ with $c \in Z_{N^2}$ and the secret key sk being the Euler-phi function $\phi(N)$ is defined by the following two steps:

1. Compute $\hat{m} = \frac{\left[\left(c^{\phi(N)} \bmod N^2\right)-1\right]}{N}$ over the integer field;

2. $Dec_{sk}(c) = \hat{m} \cdot \phi(N)^{-1} \bmod N$

The Paillier system is secure under the decisional composite residuosity assumption and we refer interested readers to [49, ch.11] for details. Paillier is homomorphic over addition in Z_N and the corresponding function is multiplication over the ciphertext field Z_{N^2}. We can also carry out multiplication with a known plaintext in the

encrypted domain. These properties are summarized below:

$$Enc_{pk}(x+y) = Enc_{pk}(x) \cdot Enc_{pk}(y) \tag{2.8}$$

$$Enc_{pk}(xy) = Enc_{pk}(x)^y \tag{2.9}$$

Multiplication with a number to which only the ciphertext is known can also be accomplished with a simple communication protocol. Assume that Bob wants to compute $Enc_{pk}(xy)$ based on the ciphertexts $Enc_{pk}(x)$ and $Enc_{pk}(y)$. Alice has the secret key sk but Bob wants to keep x, y and xy hidden from Alice. MULT$(Enc_{pk}(x), Enc_{pk}(y))$ (Protocol 1) is a secure protocol that can accomplish this task. It is secure because Alice can gain no knowledge about x and y from the uniformly random $x-r$ and $y-s$ where r and s are two random numbers generated by Bob, and Bob is never exposed to any plaintext related to x and y. The complexities of MULT$(Enc_{pk}(x), Enc_{pk}(y))$ are three encryptions and seven encrypted-domain operations (multiplication and exponentiation) on Bob side, as well as two decryptions and one encryption on Alice side. The communication costs are three encrypted numbers. The homomorphic properties and this protocol will be used extensively throughout this manuscript.

2.2.2 Parallel Oblivious Transfer (OT)

A parallel 1-out-2 Oblivious Transfer for ℓ strings having bitlength t is a two-party protocol where $\mathcal{S}$ inputs ℓ pairs of t-bit strings $S_i = \langle s_i^0, s_i^1 \rangle$ for $i = 1, \ldots, \ell$ with $s_i^0, s_i^1 \in \{0,1\}^\ell$ and $\mathcal{C}$ inputs ℓ choice bits $b_i \in \{0,1\}$. At the end of the protocol, $\mathcal{C}$ learns $s_i^{b_i}$, but nothing about $s_i^{1-b_i}$ whereas $\mathcal{S}$ learns nothing about b_i. We use the OT protocol as a black-box primitive in our constructions. It can be instantiated

Protocol 1 Private Multiplication MULT($Enc_{pk}(x), Enc_{pk}(y)$)

Require: Bob: $Enc_{pk}(x)$, $Enc_{pk}(y)$; Alice: sk
Ensure: Bob computes $Enc_{pk}(xy)$

1. Bob sends $Enc_{pk}(x-r) = Enc_{pk}(x) \cdot Enc_{pk}(-r)$ and $Enc_{pk}(y-s) = Enc_{pk}(y) \cdot Enc_{pk}(-s)$ to Alice where r and s are uniformly random numbers generated by Bob.
2. Alice decrypts $Enc_{pk}(x-r)$ and $Enc_{pk}(y-s)$, computes $Enc_{pk}[(x-r)(y-s)]$ and send it to Bob.
3. Bob computes $Enc_{pk}(xy)$ in the encrypted domain as follows:

$$\begin{aligned} Enc_{pk}(xy) &= Enc_{pk}[(x-r)(y-s) + xs + yr - rs] \\ &= Enc_{pk}[(x-r)(y-s)] \cdot Enc_{pk}(x)^s \cdot Enc_{pk}(y)^r \cdot Enc_{pk}(-rs) \end{aligned}$$

efficiently with different protocols [17, 50, 51]. In this paper we consider the protocol described in [17], which - when implemented over a suitably chosen elliptic curve - has asymptotic communication complexity $6\ell t$ and is secure against malicious $\mathcal{C}$ and semi-honest $\mathcal{S}$ in the random oracle model. Extensions of [52] can be used to reduce the number of computationally expensive public-key operations to $\approx 6t^2 + 4\ell t$ and is used when $\ell > 3t$. Moreover OT can be precomputed [53], performing an offline OT on random values that is later used in the online OT phase to obtain the correct result from the actual input values with asymptotic complexity $2\ell t$ bits.

2.2.3 Garbled Circuit (GC)

Yao's Garbled Circuit approach [48, 54], excellently presented in [55], is the most efficient method for secure evaluation of a boolean circuit C in the two party setting.

We summarize the main ideas in the following. First, the circuit **constructor** (server $\mathcal{S}$), creates a *garbled circuit* $\tilde{C}$: for each wire W_i of the circuit, he randomly

chooses a *complementary garbled value* $\tilde{W}_i = \langle \tilde{w}_i^0, \tilde{w}_i^1 \rangle$ consisting of two secrets, $\tilde{w}_i^0$ and $\tilde{w}_i^1$, where $\tilde{w}_i^j$ is the *garbled value* of W_i's value j. (Note: $\tilde{w}_i^j$ does not reveal j.) Further, for each gate G_i, $\mathcal{S}$ creates and sends to the **evaluator** (client $\mathcal{C}$) a *garbled table* $\tilde{T}_i$ with the following property: given a set of garbled values of G_i's inputs, $\tilde{T}_i$ allows to recover the garbled value of the corresponding G_i's output, and nothing else. Then garbled values corresponding to $\mathcal{C}$'s inputs x_j are (obliviously) transferred to $\mathcal{C}$ with a parallel oblivious transfer protocol: $\mathcal{S}$ inputs complementary garbled values $\tilde{W}_j$ into the protocol; $\mathcal{C}$ inputs x_j and obtains $\tilde{w}_j^{x_j}$ as outputs. Now, $\mathcal{C}$ can evaluate the garbled circuit $\tilde{C}$ to obtain the garbled output simply by evaluating the garbled circuit gate by gate, using the garbled tables $\tilde{T}_i$. Correctness of GC follows from the method of construction of the garbled tables $\tilde{T}_i$.

Here is an example of using GC to evaluate an **AND** circuit with two 1-bit inputs. Table 2.1 lists all possible results with the input variables. Then $\mathcal{S}$ constructs garbled circuit $\tilde{C}$ which includes all garbled values, while $\tilde{w}_{\mathcal{S}}^0$ & $\tilde{w}_{\mathcal{S}}^1$ are for $\mathcal{S}$ itself and $\tilde{w}_{\mathcal{C}}^0$ & $\tilde{w}_{\mathcal{C}}^1$ are for $\mathcal{C}$; and the garbled table $\tilde{T}$ as shown in Table 2.2. Suppose TSS has the secret value 0 while $\mathcal{C}$ has secret value 1, the evaluation of GC is worked as follows: 1) $\mathcal{S}$ sends the output column of Table 2.2 to $\mathcal{C}$ in a random permuted order. 2) $\mathcal{S}$ sends $\tilde{w}_{\mathcal{S}}^0$ directly to $\mathcal{C}$. Since all wire values are generated randomly and independent, $\mathcal{C}$ cannot know if $\tilde{w}_{\mathcal{S}}^0$ corresponds to 0 or 1. 3) $\tilde{w}_{\mathcal{C}}^1$ is sent to $\mathcal{C}$ via Oblivious Transfer (OT) I introduced in Section 2.2.2, in this way, $\mathcal{S}$ cannot know which secret value that $\mathcal{C}$ selects. 4) $\mathcal{C}$ uses the two wire values ($\tilde{w}_{\mathcal{S}}^0$, $\tilde{w}_{\mathcal{C}}^1$) he received to decrypted all output values in Table 2.2 and only the second row can be correctly decrypted.

High-Speed evaluation of GC [56] is feasible by using a cryptographic hash function

Table 2.1: Truth Table of AND Circuit (0:False; 1:True)

	Input		output
Row #	$\mathcal{S}$	$\mathcal{C}$	$\mathcal{C}$
1	0	0	0
2	0	1	0
3	1	0	0
4	1	1	1

Table 2.2: Garbled Table of AND Circuit

	Input		output
Row #	$\mathcal{S}$	$\mathcal{C}$	$\mathcal{C}$
1	$\tilde{w}_{\mathcal{S}}^0$	$\tilde{w}_{\mathcal{C}}^0$	$Enc_{\tilde{w}_{\mathcal{S}}^0,\tilde{w}_{\mathcal{C}}^0}(0)$
2	$\tilde{w}_{\mathcal{S}}^0$	$\tilde{w}_{\mathcal{C}}^1$	$Enc_{\tilde{w}_{\mathcal{S}}^0,\tilde{w}_{\mathcal{C}}^1}(0)$
3	$\tilde{w}_{\mathcal{S}}^1$	$\tilde{w}_{\mathcal{C}}^0$	$Enc_{\tilde{w}_{\mathcal{S}}^1,\tilde{w}_{\mathcal{C}}^0}(0)$
4	$\tilde{w}_{\mathcal{S}}^1$	$\tilde{w}_{\mathcal{C}}^1$	$Enc_{\tilde{w}_{\mathcal{S}}^1,\tilde{w}_{\mathcal{C}}^1}(1)$

$H(\cdot)$ (chosen from the SHA-2 family). The creation of the garbled table associated to a d-input gate requires 2^d invocations of $H(\cdot)$. A *point-and-permute technique* can be used to speed up the implementation of the GC protocol [56]: the garbled values $\tilde{w}_i = \langle k_i, \pi_i \rangle$ consist of a symmetric key $k_i \in \{0,1\}^t$ and $\pi_i \in \{0,1\}$ is a random permutation bit. The permutation bit π_i is used to select the right table entry for decryption with the key k_i, hence only one invocation of $H()$ for each table is needed during evaluation. The *free-XOR* gates technique introduced in [57], can be used to further improve the performance of the GC technique, so that XOR gates need not be created nor their corresponding garbled tables transmitted and evaluation is performed by a simple XOR operation. The output of the GC is converted to plain values by using a two rows conversion table for each output bit.

2.3 Information Theoretic SMC (IT-SMC)

In CS-SMC, private information is first encrypted or transformed before transmitting to other parties. The security is based on the computational burden of performing the inverse transformation such as factorization of large primes or performing discrete logarithm. To ensure even a short-term protection against adversaries, a large security parameter needs to be used which results in a hundred to a thousand-fold increase in data size, which is the weakness of CS-SMC and need to be improved in my proposed system. On the contrary, in Information Theoretic SMC (IT-SMC), no matter how computationally powerful the adversary is, an adversary will learn nothing about the secret numbers of the honest parties. The idea is that while the adversary may control a number of parties who receives messages from other honest senders, these messages provide no useful information about the secret numbers of the senders [14].

If only two parties are involved in the proposed algorithm, CS-SMC is a better choice since IT-SMC needs at least three parties. However, the proliferation of cloud-based distributed computing make possible the joint computation of secure function evaluation with the untrusted third party. The third party are only assisting the computations without any private information to be protected.

IT-SMC protocols protect privacy in such a way that the information exchanged in the protocol provides no additional information, measured in entropy, about the private data. A major disadvantage of IT-SMC, however, is the need to maintain multiple non-colluding computing parties [58]. Here I will introduce one of main primitives of IT-SMC: Shamir's Secret Share (SSS) [59].

2.3.1 Shamir's Secret Share (SSS)

Let x be a number in a finite field F_m. Let n be the number of parties and t, called threshold, be a positive integer between 1 and n. A (t, n) secret-sharing scheme of a secret number x produces n shares $[x]_i^{m,t}, i = 1, 2, \ldots, n$ such that any group of t or more shares can be used to reconstruct x. Any group of less than t shares, however, provide no information about x. The Shamir's Secret Sharing scheme hides the secret as the constant term of a random $(t-1)^{th}$ polynomial and generates the i^{th} share by evaluating the polynomial at a public constant k_i:

$$[x]_i^{m,t} \triangleq \sum_{j=1}^{t-1} \alpha_j k_i^j + x \bmod m \tag{2.10}$$

where α_j's are uniformly random numbers selected by the secret owner. The fact that α_j's are uniformly random in a finite field implies that the shares must also be uniformly random, thereby providing no information about x. Given at least t shares, the secret number x can be reconstructed with Lagrange interpolation[1]:

$$x = \sum_{i \in K} \gamma_i [x]_i^m \bmod m \tag{2.11}$$

where $\gamma_i \triangleq \prod_{1 \le j \le n, j \ne i} \frac{-k_j}{k_i - k_j}$ and K is any subset of $\{1, \ldots, n\}$ with at least t elements.

Let $x, y \in F_m$ be secret numbers and $a, b \in F_m$ be constants. The following properties of Shamir's scheme are well known [60]:

(P1) $[x + a \bmod m]_i^m = [x]_i^m + a \bmod m$

(P2) $[ax \bmod m]_i^m = a[x]_i^m \bmod m$

[1]To simplify the notations, the superscript t in $[x]_i^{m,t}$ shall be omitted if it is not affected by the operations.

(P3) $[x + y \bmod m]_i^{m,\max(s,t)} = [x]_i^{m,s} + [y]_i^{m,t} \bmod m$

(P4) $[xy \bmod m]_i^{m,(s+t)-1} = [x]_i^{m,s}[y]_i^{m,t} \bmod m$

(P5) Assume $x, y \in \{0, 1\}$ and $\oplus$ denotes xor.

$$[x \oplus y]_i^{m,(s+t)-1} = [x]_i^{m,s} + [y]_i^{m,t} - 2[x]_i^{m,s}[y]_i^{m,t} \bmod m$$

(P6) $[x]_i^{m,t} = \sum_{j=1}^{n} \gamma_j [[x]_j^{m,(s+t)-1}]_i^{m,t} \bmod m$

P1 through P5 form the foundation of computation in secret shares – they show that performing certain operations on each share of secret numbers is equivalent to applying those operations first on the secret numbers and then creating the shares. These operations include addition and multiplication with constants, with other secret numbers, and exclusive-or on secret bits. These operations are universal in the sense that any computation on a digital computer can be composed by successive applications of these fundamental operations. Since the original shares do not reveal any information about the secret numbers, no successive operations on the shares can gain further knowledge. At the end of the operations, the secret owner can collect enough number of shares to reveal the result.

Multiplication and exclusive-or operations (P4 and P5) produce results in a sharing scheme with a higher threshold $(s+t)-1$, where s and t are the original threshold of the two secret operands. Repeated applications of such operations will eventually arrive at a threshold larger than the number of parties n and the final result cannot be reconstructed even if all the shares are available. One intuitive method to solve this problem is to increase the number of parties n to guarantee that n is larger than

all threshold values in all following computation. Suppose there are η multiplication and exclusive-or operations in all computations and 2 shares of secret numbers can recover the secret at the beginning, $\eta + 2$ parties will be needed and $\eta + 1$ operands are broken into $\eta + 2$ shares and sent to the corresponding parties if the final secret result is recovered without the communication between computing agents. Therefore, the communication complexity is

$$D = (\eta + 2)(\eta + 1) \log m + (\eta + 2) \log m = (\eta + 2)^2 \log m \tag{2.12}$$

where $\eta + 2$ parties send one share out for recovering the final result.

Another solution is to renormalize the threshold using P6 so that n is independent of the number of multiplication and exclusive-or operations: each party further breaks its share into separate shares and sends each regenerated share to its corresponding party. The final share at each party is computed by a weighted summation of these newly received shares from other parties. Since each party receives only one share from any other party, no secret information is leaked. There are however hidden communication cost associated with some of these operations. This threshold reduction requires $n(n-1) \log m$ bits to be exchanged among the n parties. Suppose there are only 3 parties, which is the least number of computing parties needed for recovering one production, and η multiplication or exclusive-or operations are computed, the threshold must be reduced to 3 finally. Since the original threshold is 2, each time multiplication or exclusive-or is computed, the threshold is increased by 1 and need to be reduced using P6 with $6 \log m$ bits to be exchanged and there are $\eta - 1$ threshold reductions in total and the communication complexity of this solution is

$$D' = 6(\eta - 1)\log m + 3(\eta + 1)\log m + 3\log m = 9\eta \log m \tag{2.13}$$

Compare D' with D, the first solution has less communication complexity when $\eta \leq 4$.

To highlight the communication, I use the notation

(P6') $P_j : [[\text{expr}(x)]_j]_i^m \longrightarrow P_i : [x]_i^m$ for $i \neq j$

The expr operator in P6' can include a composition of different operations that may result in one or more steps of renormalization.

An obvious omission from the above properties is division between two secret numbers. To compute $xy^{-1} \bmod m$, y^{-1} must exists in F_m. We denote the inverse operation as follows:

(P7) INVERSE $([y]_i^m$ all $i) \longrightarrow P_i : ([y^{-1}]_i^m)$

INVERSE can be implemented by repeated multiplications according to the Carmichael's theorem [61]: $y^{-1} = y^{\lambda(m)-1} \bmod m$, where $\lambda(m)$ is the (reduced) totient function. Notice that with this equation, the inverse of 0 is defined and is equal to 0. For prime m, $\lambda(m) = m - 1$. For large $\lambda(m) - 1$, the inverse operation is expensive as every multiplication requires a renormalization step. To reduce the number of multiplications, we can first express $\lambda(m) - 1$ as a sum of powers of two, say $\lambda(m) - 1 = 1011$ base 2 which implies $y^{\lambda(m)-1} = y^4 y^2 y$. We can then recursively compute $[y^2]_i^{t,m}$ and $[y^4]_i^{t,m}$ before multiplying them together to get the final answer. The communication complexity will be $O(\log \lambda(m))$ rather than $O(\lambda(m))$ in the sequential multiplication.

Chapter 3

Related work

The main contributions of my dissertation are the introduction of the ABAC concept and a practical design of such a system using iris biometrics. The previous techniques for preserving the privacy of biometric data while maintaining their usability in various scenarios are introduced in Section 3.1. The main hurdle in applying computationally-secure SMC protocols to ABAC and the possible cryptographic primitives are presented in Section 3.2. Finally, earlier works in managing the privacy information in privacy-aware environments and the approaches to combine it with the anonymity of users' biometric signals are exhibited in Section 3.3.

3.1 Biometric and Privacy

There are other work that deal with the privacy and security issues in biometric systems but their focus are different from this dissertation. A privacy-protecting technology called "Cancelable Biometrics" has been proposed in [62]. To protect the security of the raw biometric signals, a cancelable biometric system distorts a biometric signal using a specially designed non-invertible transform so that similarity comparison can still be performed after distortion. Biometric Encryption (BE) described in [63] possesses all the functionality of Cancelable Biometrics, and is immune

against the substitution attack because it outputs a key which is securely bound to a biometric. The BE templates stored in the gallery have been shown to protect both the biometrics themselves and the keys. The stored BE template is also called "helper data". "Helper data" is also used in [33] to assist in aligning a probe with the template that is available only in the transformed domain and does not reveal any information about the fingerprint.

All the above technologies focus on the security and privacy of the biometric signals in the gallery: instead of storing the original biometric signal, they keep only the transformed and non-invertible feature or helper data extracted from the original signal that do not compromise the security of the system even if they are stolen. In these systems, the identity of the user is always recognized by the system after the biometric matching is performed. To the best of our knowledge, there are no other biometric access systems that can provide access control and yet keep the user anonymous using iris patterns. Though our focus is on user anonymity, our design is complementary to cancelable biometrics and it is conceivable to combine features from both types of systems to achieve both data security and user anonymity.

Anonymity in biometric features like faces is considered in [64]. Face images are obfuscated by a face de-identification algorithm in such a way that any face recognition softwares will not be able to reliably recognize de-identified faces. The model used in [64] is the celebrated k-anonymity model which states that any pattern matching algorithm cannot differentiate an entry in a large dataset from at least $k-1$ other entries [65, 34]. The k-anonymity model is designed for data disclosure protocols and cannot be used for biometric matching for a number of reasons. First, despite

the goal of keeping the user anonymous, it is very important of an ABAC system to verify that a user is indeed in the system. Face de-identification techniques provide no guarantee that only faces in the original database will match the de-identified ones. As such, an imposter may gain access by sending an image that is close to an de-identified face. Second, de-identification techniques group similar faces together to facilitate the public disclosure of the data. This is detrimental to anonymity as face clusters may reveal important identity traits like skin color, facial structure, etc.

3.2 Complexity Reduction in SMC

Another key difference between anonymity in data disclosure and biometric matching is the need for secure collaboration between two parties – the biometric server and the user. The formal study of such a problem is Secure Multi-party Computation (SMC). SMC is one of the most active research areas in cryptography and has wide applications in electronic voting, online bidding, keyword search and anonymous routing. Moreover, many of the basic components in a BAC system can be made secure under this paradigm. They include inner product [66, 67], polynomial evaluation [68, 69, 20], thresholding [70, 71, 48], median [16], matrix computation [72, 73], logical manipulation [74], k-means clustering [75, 76], decision tree [77, 78, 79] and other classifiers [80, 81, 69, 82] etc. A recent tutorial in SMC for signal processing community can be found in [83].

The main hurdle in applying computationally-secure SMC protocols to biometric matching is their high computational complexity. For example, the classical solution

to the thresholding problem[1], or comparing two private numbers a and b, is to use Oblivious Transfer (OT) [21]. OT is a SMC protocol for joint table lookup. The privacy of the function is guaranteed by having the entire table encrypted by a pre-computed set of public keys and transmitted to the other party. The privacy of the selection of the table entry is protected based on obfuscating the correct public key among the dummy ones. Even with recent advances in reducing the computational and communication complexity [15, 16, 17, 18, 19, 20], the large table size, the intensive encryption and decryption operations render OT difficult for pixel or sample-level signal processing operations.

A faster but less general approach is to use Homomorphic Encryption (HE) which preserves certain operations in the encrypted domain [47]. Recently, the homomorphic encryption scheme proposed by IBM and Stanford researcher C. Gentry has generated a great deal of excitement in using HE for encrypted domain processing [84]. He proposed using Ideal Lattices to develop a homomorphic encryption system that can preserve both addition and multiplication operations. This solves an open problem on whether there exists a semantically-secure homomorphic encryption system that can preserve both addition and multiplication. On the other hand, his construction is based on protecting the simplest boolean circuit and its generalization to realistic application is questionable. In an interview, Gentry estimates that performing a Google search with encrypted keywords would increase the amount of computing time by about a trillion [85] and even this claim is already challenged by others to be too conservative [86].

[1]This problem is commonly referred to as the Secure Millionaire Problem in SMC literature.

More practical homomorphic encryptions such as Paillier cryptosystem can only support addition between two encrypted numbers, but do so over a much larger additive plaintext group, thus providing a wide dynamic range for computation [87]. Furthermore, as illustrated in Section 2.2.1, multiplication between encrypted numbers can be accomplished by randomization and interaction between parties. Recently, Paillier encryption is being applied in a number of fundamental signal processing building blocks [88] including basic classifiers [81] and Discrete Cosine Transform [89] in encrypted domain. Nevertheless, the public-key encryption and decryption processes in any homomorphic encryption still pose a formidable complexity hurdle to overcome. For example, the fastest thresholding result takes around 5 seconds to compare two 32-bit numbers using a modified Paillier encryption system with a key size of 1024 bits [70]. One of the goals of this dissertation to utilize homomorphic encryption to construct a realistic biometric matching system that can tradeoff computation complexity with user anonymity in a provably secure fashion.

Another prevailing approach of implementing SMC protocol is to use Garbled Circuits (GC) [48]. GC provides a generic implementation of any binary function by having one party prepared an encrypted boolean circuit, and another party obliviously evaluated the circuit without access to intermediate values. While HE is very efficient for large integer fields, iris matching consists of mostly binary operations and is conceptually more suitable for GC. Blanton et al. proposed a hybrid approach of GC and HE for iriscode and achieved a more efficient implementation [31] than using HE alone. Recent research efforts have significantly improved the efficiency of GC [57, 90]. GC is likely to become a more efficient alternative than HE as GC

theory relies almost exclusively on symmetric encryption and HE on asymmetric encryption. Furthermore, GC is characterized by shorter security parameters, which become more pronounced when we pass from short term to medium term and long term security [91]. As such, it is attractive to develop the iriscode matching by using only GC. In this dissertation, I demonstrate a computationally efficient GC-based iriscode matching algorithm. A novel contribution is the adoption of a simplified masking technique for iriscode which significantly reduces the complexity of the circuit.

3.3 Privacy Information Management (PIM) in Video Surveillance Network

To tackle the problem in managing privacy information, earlier work like [92] introduces a framework which advocates the presence of a trusted middleware agent, referred to as Discreet Box in [92]. The Discreet Box acts as a three way mediator between the law, the users and the service providers. This centralized unit acts as a communication point between various parties and enforces the privacy regulations. Fidaleo et al. describe a secure sharing scheme in which the surveillance data is stored in a centralized server core [93] . A Privacy buffer zone, adjoining the central core, manages the access to this secure area by filtering appropriate personally identifiable information thereby protecting the data. Both approaches adopt a centralized management of privacy information making them vulnerable to concerted attacks. Cheung et al. propose a new management system within which the users and the client agents can anonymously exchange data, credential, and authorization

information [94]. This approach is reminiscent to a Data Right Management (DRM) system where the content owner can control the access of his/her content after proper payment is received. A trusted mediator agent is still required to ensure that the user and the client agents can anonymously exchange data request, credential and authorization. Furthermore, anonymous access control is not implemented in the system and the mediator can associate the encrypted videos with the identity of an individual. The PIM system proposed in this dissertation has the advantage that no trusted mediator is required.

As alluded in Chapter 1, one approach to combine anonymity in biometric access control and privacy data management is to encrypt the privacy information using the biometric signal itself. Methods that use biometric to protect sensitive data are referred to as biometric cryptosystems [95]. They have been applied in a number of practical biometric systems [96, 97, 33, 98] in which a random key is protected by a biometric signal to produce a privacy template [96, 97] or helper data [33, 98]. Such a privacy template or helper data can only be decrypted by another biometric sample from the same individual. The purpose of their proposed protocols is to protect the security of the biometric system against the attack to central server by replacing the raw biometric samples with these templates. For our application, we use biometric cryptosystems to protect the AES keys that encrypt the privacy imagery. In [96], a key-binding iris template scheme is proposed that relies on error correction coding (ECC) coding to cope with small variations between different iris patterns from the same individual. A concatenated-coding scheme is adopted to correct two types of errors: random errors brought by CCD camera pixel noise and iris distortion are

corrected by Hadamard code while burst error introduced by undetected eyelashes and specular reflections are corrected by Reed-Solomon code. While ECC-based techniques are efficient, its nature of bit error correction dictates the use of hamming distance in measuring similarity between biometric signals and limits the choice threshold tolerance in the matching process. The proposed approach in my dissertation uses a general GC-based SMC protocol which is capable of arbitrary complex matching protocols and arbitrary choice of error tolerance.

Chapter 4

SMC-based Anonymous Biometric Matching

In this chapter, I propose an implementation of an ABAC system on iris biometrics that is robust under semi-honest security model with the secure collaboration between two parties. The procedure is based on two popular computationally-secure SMC (CS-SMC) protocols, namely Homomorphic Encryption (HE) and Garbled Circuits (GC). Moreover, I explore the possibility of using Information Theoretic SMC (IT-SMC) protocols, mainly with Shamir's Secret Sharing (SSS) as the building blocks of ABAC.

4.1 Homomorphic Encryption based ABAC

In this section, I describe the implementation of an ABAC system on iris features using homomorphic encryption. The system consists of three main steps: distance computation, bit extraction and secure comparison. Except for the first step of distance computation which is specific towards iris comparison, the remaining two steps and the overall protocol are general enough for other types of biometric features and similarity search. I shall follow a bottom-up approach by first describing individual components and demonstrating their safety before assembling them together as an ABAC system.

4.1.1 Hamming Distance

The modified Hamming distance $d_H(\mathbf{x}, \mathbf{y})$ described in Equation (2.2) is used to measure the dissimilarity between iris patterns $\mathbf{x}$ and $\mathbf{y}$ which are both 9600 bits long [99]. As the division in Equation (2.2) may introduce floating point numbers, we focus on the following distance and roll the denominator into the similarity threshold during the later stage of comparison.

$$\widehat{d_H}(\mathbf{x}, \mathbf{y})^2 := \| \ (\mathbf{x} \otimes \mathbf{y}) \cap mask_{\mathbf{x}} \cap mask_{\mathbf{y}} \ \|_2^2 \tag{4.1}$$

DIST (Protocol 2) provides a secure computation of the modified Hamming distances between Alice's probe $\mathbf{q}$ and Bob's DB. Alice needs to provide the encryption of individual bits $\mathbf{q} = (q_1, q_2, \ldots, q_n)^T$ and their negation to Bob. Even though Bob can compute the negation in the encryption domain by performing $Enc_{pk}(\neg q_i) = Enc_{pk}(1 - q_i) = Enc_{pk}(1) \cdot Enc_{pk}(q_i)^{-1}$, it is computationally more efficient for Alice to compute them in plaintext as demonstrated in Section 4.4.1. In step 1a, Bob computes the XOR between each bit of the query and the corresponding bit in each record $\mathbf{x}_i$. $\widehat{d_H}(\mathbf{q}, \mathbf{x}_i)$ can then be computed by summing all the XOR results in the encrypted domain. Bob cannot derive any information about Alice's probe as the operations are all performed in the encrypted domain. Alice does not participate in this protocol at all. The complexity of DIST includes $O(Nn)$ encrypted-domain operations where N is the size of DB and n is the number of bits for each feature vector.

Protocol 2 Secure computation of distances DIST($DB, Enc_{pk}(q_j)$, $Enc_{pk}(\neg q_j)$)

Require: Bob: $\mathbf{x}_i$ for $i = 1, \ldots, N$, $Enc_{pk}(q_j)$ and $Enc_{pk}(\neg q_j)$ for $j = 1, \ldots, n$

Ensure: Bob computes $Enc_{pk}\left[\widehat{d_H}(\mathbf{q}, \mathbf{x}_i)^2\right]$ for $i = 1, \ldots, N$.

1. For $i = 1, \ldots, N$, Bob repeats the following two steps:

   a) For $k = 1, \ldots, n$, compute

   $$Enc_{pk}(q_k \otimes x_k^i) = \begin{cases} Enc_{pk}(q_k) & \text{if } x_k^i = 0, \\ Enc_{pk}(\neg q_k) & \text{otherwise} \end{cases}$$

   b) Compute

   $$\begin{aligned} Enc_{pk}\left[\widehat{d_H}(\mathbf{q}, \mathbf{x}_i)^2\right] &= Enc_{pk}\left(\sum_{k:\left[mask_{\mathbf{q}} \cap mask_{\mathbf{x}_i}\right]_i = 1} q_k \otimes x_k^i\right) \\ &= \prod_{k:\left[mask_{\mathbf{q}} \cap mask_{\mathbf{x}_i}\right]_i = 1} Enc_{pk}(q_k \otimes x_k^i) \end{aligned}$$

4.1.2 Bit Extraction

The next step is to compare the calculated encrypted distance with a plaintext threshold. As comparison cannot be expressed in terms of summation and multiplication of the two numbers, we need to first extract individual bits from the encrypted distance. EXTRACT($Enc_{pk}(x)$) (Protocol 3) is a secure protocol between Bob and Alice to extract individual encrypted bits $Enc_{pk}(x_k)$ for $k = 1, \ldots, l$ from $Enc_{pk}(x)$ where x is a l-bit number. The idea is for Bob to ask Alice's assistance in decrypting the numbers and extracting the bits. To protect Alice from knowing anything about x, Bob sends $Enc_{pk}(x + r)$ to Alice who then extracts and encrypts individual bits $Enc_{pk}\left[(x + r)_k\right]$. Except for the least significant bit (LSB), Bob cannot undo the randomization in $Enc_{pk}\left[(x + r)_k\right]$ by carrying out an XOR operation with the bits

Protocol 3 Bit Extraction EXTRACT($Enc_{pk}(x)$)

Require: Bob: $Enc_{pk}(x)$ where x is a l-bit number; Alice sk.
Ensure: Bob computes $Enc_{pk}(x_k)$ for $k = 1, \ldots, l$ with $k = 1$ being the LSB.

1. Bob creates a temporary variable $Enc_{pk}(y) := Enc_{pk}(x)$.
2. For $k = 1, \ldots, l$, the following steps are repeated
   a) Bob generates a random number r and sends $Enc_{pk}(y + r)$ to Alice.
   b) Alice decrypts $y+r$, extracts the k^{th} bit $(y+r)_k$ and sends $Enc_{pk}[(y + r)_k]$ back to Bob.
   c) Bob computes $Enc_{pk}(x_k) := Enc_{pk}[(y + r)_k \otimes r_k]$.
   d) Bob updates $Enc_{pk}(y) := Enc_{pk}(y - x_k 2^{k-1}) = Enc_{pk}(y) \cdot Enc_{pk}(x_k)^{-2^{k-1}}$

of r due to the carry bits. To rectify this problem, step 2d in EXTRACT zeros out the lower order bits after they have been extracted and stores the intermediate result in y, thus guaranteing the absence of any carry bits from the lower order bits during the randomization. Alice cannot learn any information about y because the bit to be extracted, $(y + r)_k$, is uniformly distributed between 0 and 1. Plaintexts obtained by Alice in different iterations are also uncorrelated as a different random number is used by Bob in each iteration. Even though Alice wants to make x as small as possible to pass the comparison test, there is no advantage of replacing her replies to Bob with any other value. Bob is not able to obtain any information about x either as all operations are performed in the encrypted domain. Based on the security model introduced in Section 2.1.3, this protocol is secure. The complexities of EXTRACT are l encryptions and $O(l)$ encrypted-domain operation for Bob, as well as l decryptions and l encryptions for Alice. The communication costs are $2l$ encrypted numbers.

4.1.3 Threshold Comparison

Based on the encrypted bit representations of the distances, we can carry out the actual threshold comparison. COMPARE($Enc_{pk}(x_k), y_k$ for $k = 1, \ldots, l$) (Protocol 4) is based on the secure comparison protocol developed in [100]. Step 2a accumulates the differences between the two numbers starting from the most significant bits. The state variable $w = 0$ at the k^{th} step implies that the bits at order k and higher between x and y match perfectly with each other. Step 2b then computes $Enc_{pk}(c_k)$ where $c_k = 0$ if and only if $w = 0$, $x_k = 0$ and $y_k = 1$. This implies that $x < y$. In other words, $x < y$ is true if and only if there exists $c_k = 0$. In the last step, we invoke the secure multiplication as described in Protocol 1 to combine all c_k together into c which is the desired output. Bob gains no knowledge in this protocol as he never handles any plaintext data. The only step that Alice involves in is in the secure multiplication. The adversarial intention of Alice is to make c zero so as to pass the comparison test. However, the randomization step in Protocol 1 provides no additional knowledge nor advantage for Alice to change her input. Thus, this protocol is secure. The complexities of COMPARE are $3l$ encryptions and $O(l)$ encrypted-domain operations on Bob side, as well as $2l$ decryptions and l encryptions on Alice side. The communication costs are $3l$ encrypted numbers.

4.1.4 Overall Algorithm

Protocol 5 defines the overall ABAC system. Steps 1 and 2 show that Alice first sends Bob her public key and the encrypted bits of her probe. Steps 3 and 4 use

Protocol 4 Secure comparison COMPARE($Enc_{pk}(x_k), y_k$ for $k = 1, \ldots, l$)

Require Bob: $Enc_{pk}(x_k), Enc_{pk}(y_k)$ and y_k for $k = 1, \ldots, l$; Alice: sk
Ensure Bob computes $Enc_{pk}(c)$ such that $c = 0$ if $x < y$.

1. Bob sets $Enc_{pk}(c) := Enc_{pk}(1)$, $Enc_{pk}(w) := Enc_{pk}(0)$.
2. For $k = l, \ldots, 1$ starting from the MSB, Bob and Alice compute
   a) $Enc_{pk}(w) := Enc_{pk}[w + (x_k \otimes y_k)] = Enc_{pk}(w) \cdot Enc_{pk}(x_k \otimes y_k)$
   b) $Enc_{pk}(c_k) := Enc_{pk}(x_k - y_k + 1 + w) = Enc_{pk}(x_k) \cdot Enc_{pk}(y_k)^{-1} \cdot Enc_{pk}(1) \cdot Enc_{pk}(w)$
   c) $Enc_{pk}(c) := \text{MULT}(Enc_{pk}(c), Enc_{pk}(c_k))$.

Protocol 5 ABAC($DB, \mathbf{q}$)

Require: Bob: $\mathbf{x}_i, i = 1, \ldots, N$ and ϵ; Alice: $\mathbf{q}$
Ensure : Bob computes $y = 1$ if $\widehat{d_H}(\mathbf{q}, \mathbf{x}_i)^2 < \epsilon$ for some i and 0 otherwise

1. Alice sends pk to Bob.
2. Alice computes $Enc_{pk}(q_j)$ and $Enc_{pk}(\neg q_j)$ for $j = 1, \ldots, n$ and sends them to Bob.
3. Bob executes DIST($DB, Enc_{pk}(q_j)$, $Enc_{pk}(\neg q_j)$ for $j = 1, \ldots, n$) to obtain $Enc_{pk}\left[\widehat{d_H}(\mathbf{q}, \mathbf{x}_i)^2\right]$ for $i = 1, \ldots, N$.
4. For $i = 1, \ldots, N$, Bob and Alice execute EXTRACT$\left(Enc_{pk}\left[\widehat{d_H}(\mathbf{q}, \mathbf{x}_i)^2\right]\right)$ to obtain the binary representations $Enc_{pk}\left[\widehat{d_H}(\mathbf{q}, \mathbf{x}_i)^2_k\right]$ for $k = 1, \ldots, \lceil \log_2 n \rceil$.
5. Bob sets $Enc_{pk}(u) := Enc_{pk}(1)$.
6. For $i = 1, \ldots, M$, Bob and Alice computes
   a) $Enc_{pk}(c) := \text{COMPARE}(Enc_{pk}\left[\widehat{d_H}(\mathbf{q}, \mathbf{x}_i)^2_k\right], (\epsilon \| mask_{\mathbf{q}} \cap mask_{\mathbf{x}_i} \|_2^2)_k$ for $k = 1, \ldots, \lceil \log_2 n \rceil)$
   b) $Enc_{pk}(u) := \text{MULT}(Enc_{pk}(u), Enc_{pk}(c))$
7. Bob generates a random number r, computes $\text{HASH}_{pk_H}(r)$ and sends Alice $Enc_{pk}(u + r)$.
8. Alice decrypts $Enc_{pk}(u + r)$, computes $\text{HASH}_{pk_H}(u + r)$ and sends it back to Bob.
9. Bob sets $y = 1$ if $\text{HASH}_{pk_H}(r) = \text{HASH}_{pk_H}(u + r)$ and 0 otherwise.

secure distance computation DIST (Protocol 2) and secure bit extraction EXTRACT (Protocol 3) to compute the encrypted bit representations of all the distances. Steps 4 and 5 then use secure comparison COMPARE (Protocol 4) and accumulate the results into $Enc_{pk}(u)$ where $u = 0$ if and only if $\widehat{d_H}(\mathbf{q}, \mathbf{x}_i)^2 < \epsilon \cdot \|mask_{\mathbf{q}} \cap mask_{\mathbf{x}_i}\|_2^2$ for some i. To determine if Alice's probe produces a match, Bob cannot simply send Alice $Enc_{pk}(u)$ for decryption as she will simply returns a zero to gain access. Instead, Bob adds a random share r and sends $Enc_{pk}(u+r)$ to Alice. The decrypted value $u+r$ cannot be sent directly to Bob for him to compute u. Unless $u = 0$, the actual value of u should not be disclosed to Bob in plaintext as it may disclose some information about the distance computations. Instead, we assume the existence of a Collision-Resistant Hash Function HASH to which Bob and Alice share the same key pk_H [49, ch.4]. Alice and Bob compute $\text{HASH}_{pk_H}(u + r)$ and $\text{HASH}_{pk_H}(r)$ respectively. As the hash function is collision resistant, their equality implies that $u = 0$ and Bob can verify that Alice's probe matches one of the entries in DB without knowing the actual value of the probe. Since Alice knows nothing about r, she cannot cheat by sending a fake hash value. The complexities of Protocol 5 are $O(N \log_2 n)$ encryptions and $O(Nn)$ encrypted-domain operations for Bob, as well as $O(N \log_2 n)$ encryptions and decryptions for Alice. The communication costs are $O(N \log_2 n)$ encrypted numbers.

4.2 Garbled Circuits based ABAC

While HE is very efficient for large integer fields, iriscode matching consists of mostly binary operations and is conceptually more suitable for GC. Recent research efforts have significantly improve the efficiency of GC [57, 90]. Moreover, GC is likely to

become a more efficient alternative than HE as GC theory relies almost exclusively on symmetric encryption and HE on asymmetric encryption. The former is characterized by shorter security parameters, which become more pronounced when we pass from short term to medium term and long term security [91]. As such, it is attractive to develop the iriscode matching by using only GC. In this section, I demonstrate a computationally efficient GC-based iriscode matching algorithm. The main innovations compared to the prior art include:

1. an iris masking technique that simplifies the operations on the encrypted data without sacrificing the recognition rate;

2. the adoption of a matching protocol based only on garbled circuits which offers longer term security over existing solutions based on homomorphic encryption or hybrid techniques.

3. The computational and communication complexity of the on-line phase of the proposed protocol is extremely low, thus opening the way to its exploitation in practical applications.

4.2.1 GC-based Iriscode Matching

In our proposed system, the biometric server, Bob, has an iris gallery which stores the iris features $\{X_1, \ldots, X_N\}$ of N members. The user, Alice, provides a probe $q = (q_1, \ldots, q_n)$ and evaluates the GC which produces a match if there exists at least an $i \in \{1, \ldots, N\}$ such that $d(q, X_i) < \epsilon$ for a similarity threshold ϵ. $d(q, X_i)$ is a modified Hamming Distance (HD) defined in Equation (2.2) without the use of

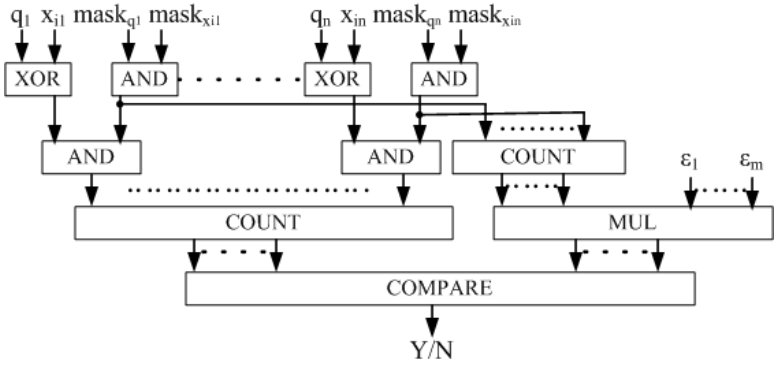
distance square.

$$d(q, X_i) := \frac{D(q, X_i)}{M(q, X_i)} = \frac{\| (q \otimes X_i) \cap mask_q \cap mask_{X_i} \|}{\| mask_q \cap mask_{X_i} \|} \tag{4.2}$$

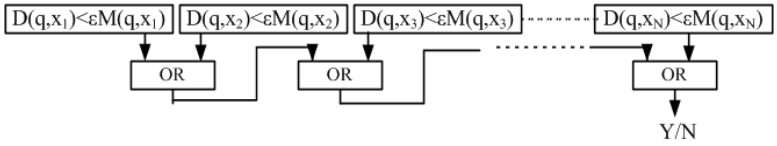
Our GC implementation has Bob first constructed a circuit that compares the input probe with the entire database and outputs the decision bit. The circuit is then sent to Alice for evaluation. Alice uses 1-out-of-2 OT to input her probe and computes the output of the circuit without learning any intermediate values. The final garbled bit is sent back to Bob for decryption. Under the semi-honest adversary model, our protocols guarantee that only Bob can know the final decision bit. The biometric probe is protected from Bob and Bob's biometric database is kept secret from Alice under any polynomial-time attacks.

Figure 4.1(a) shows the circuit for private iris-code matching between the probe q and the entry X_i in the database. It uses the basic garbled circuits (XOR, AND, and MULtiplication), a COUNT circuit to compute the number of ones in its input [101], and a COMPARE circuit to check if the first input is lower than the second input [102]. Given the fact that division in (4.2) is a complicated circuit [103] and multiplication involves fewer gates than division [104], I roll the denominator $M(q, X_i)$ of (4.2) into the similarity threshold ϵ and test whether $D(q, X_i) < \epsilon \cdot M(q, X_i)$. Since all computation should be computed over integers and ϵ is a decimal number in the range $[0, 1]$, we pre-multiply ϵ by 2^m and round it to an integer in the range $[0, 2^m]$ before taking part in the multiplication circuit with $M(q, X_i)$. Also, $D(q, X_i)$ is left shifted by m bits so the real COMPARE checks the result of $D(q, X_i) \cdot 2^m < (\epsilon \cdot 2^m) \cdot M(q, X_i)$. In order to highlight the overall structure of the circuit, I hide the scale-up processing

and use $D(q, X_i)$ and ϵ instead of $D(q, X_i) \cdot 2^m$ and $\epsilon \cdot 2^m$ in Figure 4.1(a).



(a) Sub-circuit: $D(q, X_i) < \epsilon M(q, X_i)$



(b) Whole Circuit

Figure 4.1: Circuit design for private iriscode matching

The output of the sub-circuit $D(q, X_i) < \epsilon \cdot M(q, X_i)$ cannot be made available to Bob in plaintext, otherwise, Bob will know the exact entry that matches the probe and reveal Alice's identity. In Figure 4.1(b), we use OR gates to connect the outputs of all COMPARE sub-circuits $D(q, X_i) < \epsilon \cdot M(q, X_i)$ for $i \in \{1, \ldots, N\}$ together. In the end, only the final output of all OR gates will be decoded and shared by two parties.

As described in Section 2.2.3, XOR gates can be evaluated without communication between two parties. Thus, only non-XOR gates are counted in our complexity analysis. In the sub-circuit shown in Figure 4.1(a), a substantial number of gates are devoted to incorporate individual masks in the calculation – there are n AND gates used to compare the two masks and n AND gates for the actual masking, where n is the bit-length of the iriscode. A COUNT circuit is used to aggregate the number

of non-zero common mask bits and a MUL circuit to combine the result with the similarity threshold. As such, any effort to minimize or even eliminate the variability among masks, without sacrificing the precision, can significantly reduce the complexity of the circuit. We explore the feasibility of such an approach in the next two sections.

4.2.2 Simplification of Iris Masks

Each iriscode consists of two parts: iris feature and mask. While the iris feature is confidential data, it is unclear if the mask itself contains enough sensitive information for identification. Prior schemes such as [26] make the assumption without justification that masks do not disclose identify information and are treated as public information. While such an approach can significantly reduce complexity as alluded in Section 4.2.1, there are other studies such as [105] that show eyelashes positions, which make up a significant portion of the mask, have inherent correlation and can be used to infer important ethnic information about an individual. To the best of our knowledge, the privacy leakage through iris masks has not been statistically quantified in any previous studies. Using a publicly-available iriscode database CASIA, which contains multiple iriscodes for more than 290 individuals, we statistically measure the difference between the hamming distances of iris masks for the same individuals and for different individuals. Details of the experimental results are provided in Section 4.4.2. Based on our experiments, we conclude that *iris masks from the same individual demonstrate correlations that are not present across different individuals*, and as such, *iris masks should be considered as private information at the server and*

not shared with the external biometric reader.

4.2.3 Common Mask for All Irises

Since the information of masks cannot be shared, we exploit a different approach to simplify the usage of masks. A typical mask contains information about eyelashes, eyelids, specular reflections, or other noise. We want to test the hypothesis that the positions of the noises are relatively fixed so that a common iris mask can be designed to replace the individual masks without much loss in precision. The common mask is created by ORing all the available masks in the database. Our experiments in Section 4.4.3 show that using the common mask on CASIA only results in less than 1% drop in recognition performance when compared with using individual masks. While it is our ongoing work to see if such a conclusion can be scaled up to a much larger database, we present here the design of a significantly simplified circuit which assumes that a common mask is used and publicly available.

The simplified GC sub-circuit for $D(q, X_i) < \epsilon M(q, X_i)$ is shown in Figure 4.2. We use MASK to denote the common mask and the blue-line block to highlight the gates that can be pre-computed. MASK_FILTER is a circuit that selects the parts of the iriscodes for matching according to MASK. The use of a common mask results in a speedup factor of 8.7 as demonstrated by our experiments whose details can be found in Section 4.4.

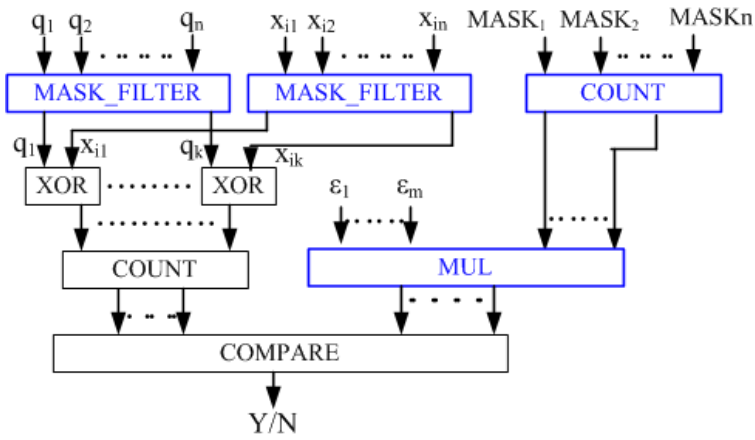


Figure 4.2: Simplified GC sub-circuit for $D(q, X_i) < \epsilon M(q, X_i)$

4.3 Secure Multi-party Computation (# of parties $\geq$ 3)

In the above sections, I have proposed two computationally-secure SMC based protocols for ABAC. In these two kinds of protocols, no matter HE or GC, only two parties are involved. Since the emergence of cloud computing make possible transferring of the computing task to oursourced companies as a third party [106], I will investigate if the participation of an additional unreliable computing party can make ABAC work under secure method.

IT-SMC introduced in Section 2.3.1 is a good way to implement the secure information exchange among more than two parties. Armed with the basic properties of Shamirs Secret Share (SSS), which is a primitives of IT-SMC, many commonly used signal processing operations can be implemented. In this section, I will implement convolution, comparison, sign function, and quantization operations in Shamir's Secret Sharing scheme. First we need to clarify its semantics in the modulo field F_m. We allow negative secret number $-x$ to be represented by $m - x$ in F_m. Thus, a

comparison of $x > 0$ is equivalent to $x \bmod m < \lceil m/2 \rceil$. Second, we need to select m to be at least *twice* as big as the dynamic range of any intermediate and final values in the target signal processing algorithm. This is necessary to ensure that we can represent both $x_{max} - x_{min}$ and $x_{min} - x_{max}$ in F_m where x_{max} and x_{min} are the largest and smallest numbers.

4.3.1 Convolution

In a linear convolution operation, two parties holding a secret signal $x(t)$ and a filter $h(t)$ can create n shares independently and distribute them to n parties to perform a privacy-protected linear filtering:

$$[conv(x,h)]_i^{m,2t-1} = \left[\sum_\tau x(t-\tau)h(\tau)\right]_i^{m,2t-1} = \sum_\tau [x(t-\tau)]_i^{m,t}[h(\tau)]_i^{m,t} \qquad (4.3)$$

Since no accumulated multiplication is computed and the threshold of recovering the convolution result is increased from 2 to 3, no renormalization is needed when 3 parties are involved. The notation of convolution function is as follows:

(P8) CONV (x at P_1, h at P_2) $\longrightarrow P_i : [conv(x,h)]_i^m$

4.3.2 Threshold comparison

Comparison is central to non-linear signal processing. To handle the non-linear nature of comparison, I rely on the radix-2 representations of the numbers. I start with a simpler version of comparison that compares two secret numbers v and w *in plaintext* stored at two different parties. To simplify the description of the protocol, we assume

a three-party computation, i.e. $n = 3$ and $t = 2$. The notation of this comparison protocol is as follows:

(P9) COMPARE2 (v at P_1, w at P_2) $\longrightarrow P_i : [v > w]_i^m$

where the binary predicate $v > w$ is 1 if true and 0 if false. The idea used for the comparison is based on the following observation: suppose v and w are l-bit operands for comparison, the Most Significant Bit (MSB) of $2^l + v - w$ is 1 if and only if $v \geq w$. We define $a \triangleq 2^l + x - y$ where a is $l + 1$-bit. The following task is to extract the MSB of a (denoted as a_l).

Protocol 6 describes the framework of the privacy-preserved comparison protocol. The first 3 steps are easy to understand based on the above observation. It is very easy to extend this protocol to the situation where only secret shares can be accessed (in this situation, start Protocol 6 from step 3). Step 4 uses the truncation protocol in [107] to cut the l bits from the left and extract MSB of the result of step 3 as the output.

Protocol 6 COMPARE2(v, w)

Require: v at P_1, w at P_2
Ensure: $[c]_i^m$ at party P_i for $i = 1, 2, 3$ where $c \triangleq (v > w)$.

1. $P_{1,2}$: $v \triangleq v_{l-1} \ldots v_0$ base 2, $w \triangleq w_{l-1} \ldots w_0$ base 2.
2. $P_{1,2}$: $[v]_i^m, [w]_i^m \rightarrow P_i$.
3. P_i: $[a]_i^m \triangleq 2^l + [v]_i^m - [w]_i^m$
4. P_i: TRUNC$([a]_i^m, l + 1, l) \rightarrow [c]_i^m$

The truncation protocol (Protocol 7) works as follows: at step 1 which denoted as the function PRandM(), the secret shares of two random numbers, r'' and r' are generated for each party. These two random values are concatenate into one $log(m)$-bit random number (denoted as r) where r'' is the first $(log(m) - l)$-bit and r' is the last l-bit of r. Therefore, $r' = r \bmod 2^l$. The function PRandM() at step 1 can be realized by an independent party who is isolated from any other operations excepting generating random numbers. This party has only the function of sending data and it cannot receive any data. Once the shares were generated, the original random value would be destroyed immediately. Another method to implement the function PRandM() can be found in [107] without an additional party.

At step 2 of the truncation protocol , each party calculates the share of $a + r$, broadcasts it to reconstruct $a + r$ which can be known by all parties. The task of reconstruction can be handed over to either party (say p_k) and it will send $a+r \bmod 2^l$ to each party. Step 4 use function BitLT (Protocol 8) to compare if $a + r \bmod 2^l < r \bmod 2^l$. This comparison is used to decide if a modulus 2^l should be added into $a + r \bmod 2^l - r \bmod 2^l$ to make the result fit in the modulo field F_{2^l}

At step 5, the secret shares of last l bits of a are generated. Subtract it from a and we can get the secret share of MSB of a finally.

The first operand of BitLT() at step 2 of Protocol 7 is a number which is available for all parties. We only need to protect the privacy of the other operand, – each party owns only a share of the second operand. Most of bit-wise operation in BitLT() are implemented without interaction (step 1, 3, 4, 5). The remaining steps only need constant-round interaction. I omit the correctness of BitLT() and the interested

Protocol 7 TRUNC$([a]_i^m, l+1, l)$

Require: $[a]_i^m$ at party P_i where $a \triangleq a_l \dots a_0$.
Ensure: $[a_l]_i^m$ at party P_i for $i = 1, 2, 3$.

1. P_i: PRandM$(log(m), l) \rightarrow ([r'']_i^m, [r']_i^m, [r'_{l-1}]_i^m, \dots, [r'_0]_i^m)$
2. P_i: output$([\mathrm{a}]_{\mathrm{i}}^{\mathrm{m}} + 2^{\mathrm{l}}[\mathrm{r''}]_{\mathrm{i}}^{\mathrm{m}} + [\mathrm{r'}]_{\mathrm{i,3}}^{\mathrm{m}}) \rightarrow b$.
3. P_k: $b' \triangleq b \bmod 2^l$, $b' \rightarrow P_i$
4. P_i: BitLT$(b', ([r'_{l-1}]_i^m, \dots, [r'_0]_i^m)) \rightarrow [u]_i^m$
5. P_i: $[a']_i^m \triangleq b' - [r']_i^m + 2^l[u]_i^m$
6. P_i: $[a_l]_i^m \triangleq ([a]_i^m - [a']_i^m)(2^{-l} \bmod m)$

Protocol 8 BitLT$(b', ([r'_{l-1}]_i^m, \dots, [r'_0]_i^m))$

Require: $b', ([r'_{l-1}]_i^m, \dots, [r'_0]_i^m)$ at party P_i where both b' and r' have length m.
Ensure: $[u]_i^m$ at party P_i for $i = 1, 2, 3$ where $u \triangleq (b' > r')$.

1. P_i: for $j = 0$ to $l-1$, do $[d_j]_i^m \triangleq b'_j \otimes [r'_j]_i^m$.
2. P_i: PreMulC$([d_{l-1}]_i^m + 1, \dots, [d_0]_i^m + 1) \rightarrow ([p_{l-1}]_i^m, \dots, [p_0]_i^m)$
3. P_i: for $j = 0$ to $l-2$, do $[s_j]_i^m \triangleq [p_j]_i^m - [p_{j+1}]_i^m$
4. P_i: $[s_k]_i^m \triangleq [p_k]_i^m - 1$
5. P_i: $[s]_i^m \triangleq \sum_{j=0}^{l-1} [s_j]_i^m (1 - b_j)$
6. P_i: $[u]_i^m \triangleq$ Mod2$([s]_i^m, l)$

reader can refer to [107] for detail. Also, the reader can find more information about the following building blocks used in the above protocols. More detail about the comparison protocols are provided in [107].

- PRandM(n, l): all parties receive their shared random values of l-bit r' and $(n-l)$-bit r''. Also, the secret shares of each bit of r' is generated.
- Output$([a]_i^m)$: all parties broadcast their shares and reconstruct a.

- PreMulC($[d_{l-1}]_i^m, \ldots, [d_0]_i^m$): after this function, each party receives l secret shares, – $([p_{l-1}]_i^m, \ldots, [p_0]_i^m)$, where p_i is prefix multiplication of the input bits such that $p_i = \prod_{j=i}^{l-1}(d_j)$.

- Mod2($[s]_i^m, l$): all parties get their shares of the least significant bit of l-bit s.

In the above SSS-based procedures, many secret shares are from l-bit privacy number and these generated shares are $log(m)$-bit. When m is big (I choose $m = 2^{50} - 1$ in Section 4.4.5), it is a waste of communication bandwidth to use $log(m)$-bit secret shares to protect 1-bit privacy number. To reduce communication complexity in this situation, the binary operands and output are represented in F_5, which is the smallest field one can use to represent a secret among three parties. Because step 6 of the function BitLT() in Protocol 8 needs to mod 2 and if the operand mod 5 before mod 2, the parity of the operand might be changed. Therefore, after each party receive the shares in F_5, the intermediate value and final output is represented in F_{10}, which means that the computed result will mod 10 instead of mod 5 before step 6 of Protocol 8. Since the result of the function Mod2() will be secret shares of either 1 or 0, which will not change if we change the module from 10 to 5, the shares of outputs will be represented again in F_5.

Also, it is needed to move the result of Mod2() back to the larger field F_m for the following operations in Protocol 7. The following PROMOTE procedure does exactly that:

(P10) PORMOTE($[x]_i^5$ all i) $\longrightarrow P_i : [y]_i^m$

where $y \bmod m = x \bmod 5$

The full detail of the PROMOTE protocol is in Protocol 9.

Protocol 9 PROMOTE($[x]_i^5$)

Require: $[x]_i^5$ at party P_i for $i = 1, 2, 3$
Ensure: $[y]_i^m$ at party P_i for $i = 1, 2, 3$ where $y \bmod m = x \bmod 5$.

1. P_1 : a random $r \longrightarrow P_2$.
2. $P_1 : u \triangleq \gamma_1[x]_1^5 - r \longrightarrow P_3$.
3. $P_2 : v \triangleq \gamma_2[x]_2^5 + r$
4. $P_3 : w \triangleq \gamma_3[x]_3^5 + u$
5. $P_{2,3}$: if v or $w = 0$, set it to 5.
6. $P_{2,3} : [v + w]_i^5 \longrightarrow\ P_i :\ [y]_i^m$
7. $P_j : [[(-4)^{-1}(y-5)(y-10)]_{j,3}]_i^m \longrightarrow\ P_i :\ [y]_i^m$

Correctness: Steps 1 through 4 create plaintext v and w at party 2 and 3 to represent x using equation 2.11 such that

$$\begin{aligned} x &= \gamma_1[x]_1^5 + \gamma_2[x]_2^5 + \gamma_3[x]_3^5 \bmod 5 \\ &= (\gamma_2[x]_2^5 + r \bmod 5) + (\gamma_3[x]_3^5 + \gamma_1[x]_1^5 - r \bmod 5) \\ &\triangleq (v \bmod 5 + w \bmod 5) \bmod 5 \end{aligned} \tag{4.4}$$

where party 1 splits his share into a random r and $\gamma_1[x]_1^5 - r$ and distributes them to party 2 and 3 respectively. Since r is random, party 2 and 3 cannot gain any knowledge about the original share of party 1. If $x < \lceil 5/2 \rceil = 3$, then one of the following two statements must hold:

$$v \bmod 5 + w \bmod 5 < 3 \tag{4.5}$$

$$5 \leq v \bmod 5 + w \bmod 5 < 8 \tag{4.6}$$

After creating v and w such that $v + w \bmod 5 = x \bmod 5$, step 5 ensures that neither v nor w can be 0. Note that if $x = 0 \bmod 5$, then $v + w \bmod m$ can be 5 or 10. If $x = 1 \bmod 5$, then $v+w \bmod m$ must be 6. Step 6 creates shares for $y = v+w \bmod m$. Step 7 computes the shares for the expression $(-4)^{-1}(y-5)(y-10) \bmod m$ which is 0 if $y = 5$ or $10 \bmod m$, and 1 if $y = 6 \bmod m$.

4.3.3 Sign Function

Since negative number is represented as $m-x$ in F_m, to simplify the decision function of negative number, we can make $m = 2^\theta - 1$, where θ is a suitable integer to make m big enough for the all intermediate values and final result. Therefore, the range of negative number is $[\lceil m/2 \rceil, m]$, where the MSB of all numbers are 1, and the positive and zero is in $[0, \lceil m/2 \rceil - 1]$, where the MSB is 0. Protocol 10 describes the method to determine the MSB of the secret number in a share scheme, which only have one step to use Protocol TRUNC to cut off the least $\theta - 1$ bits of operand x.

Protocol 10 SIGN$([x]_i^m)$

Require: $[x]_i^m$ at party P_i for $i = 1, 2, 3$
Ensure: $[s]_i^m$ at party P_i for $i = 1, 2, 3$ where $s \triangleq (x < 0)$.

1. P_i: TRUNC$([x]_i^m, \theta, \theta - 1) \rightarrow [s]_i^m$

The notation of sign function is as follows:

(P11) SIGN $([x]_i^m) \longrightarrow P_i : [x < 0]_i^m$

4.3.4 Quantization

Another indispensable operation in signal processing is quantization. This is particular important when the computation is done in fixed point format because quantization enables non-trivial computation to be computed in fixed size F_m. Uniform quantization with interval size p can be represented as equation 4.7:

$$QUANTIZE(x) \triangleq p^{-1}(x - (x \bmod p)) \bmod m \tag{4.7}$$

The key to QUANTIZE is the modulo-p operation. We can make $p = 2^l$ so that QUANTIZE procedure can be changed to TRUNC procedure in Protocol 7, which will cut the least significant l bits of x. However, when x is negative, truncating the least significant l bits might make the quantized x less than $[\lceil m/2 \rceil, m]$ which will make it be a positive number. To handle the negative number x, the absolute value of x (denoted as $\|x\|$) will be computed first. After truncating the least significant l bits of $\|x\|$, the quantized $\|x\|$ will be change to negative value by multiplying -1.

The protocol QUANTIZE is denoted and implemented as follows:

(P12) QUANTIZE $([x]_i^m) \longrightarrow P_i : [y]_i^m$

Correctness: Steps 1 computes the shares of sign of x. Step 2 get the shares of $\|x\|$: if $x < 0$, it is represented as $m - \|x\|$ and $\|x\| = m - x$; if $x \geq 0$, $\|x\| = x$. Step 2 involves the multiplication of secret shares and need to be renormalized back to the original threshold at step 3. Step 4 truncates the least significant l bits of $\|x\| = x$. Step 5-7 recover quantized $\|x\|$, denoted as $\|f\|$, into f according to the sign of x because x and quantized x, which is f, should have the same sign: negative f is

Protocol 11 QUANTIZE($[x]_i^m$)

Require: $[x]_i^m$ at party P_i for $i = 1, 2, 3$; $p = 2^l$ where $l < \log(m)$.
Ensure: $[y]_i^m$ at party P_i for $i = 1, 2, 3$ such that $y = p^{-1}(x - (x \bmod p))$.

1. P_i: SIGN($[x]_i^m$) $\rightarrow [s]_i^m$.
2. P_i: $[s(m - 2x) + x]_i^m \rightarrow [\text{expr}(z)]_i^m$.
3. $P_j : [[\text{expr}(z)]_j]_i^m \rightarrow P_i : [z]_i^m$ for $i \neq j$.
4. P_i: TRUNC($[z]_i^m, \log(m), l$) $\rightarrow [d]_i^m$
5. P_i: $[-2sd]_i^m \rightarrow [\text{expr}(f)]_i^m$.
6. $P_j : [[\text{expr}(f)]_j]_i^m \rightarrow P_i : [f]_i^m$ for $i \neq j$.
7. P_i: $[f + d]_i^m \rightarrow [y]_i^m$.

$-\|f\|$ and non-negative f is the same as $\|f\|$. Also, there are one multiplication at step 5, so step 6 renormalized the threshold back using P6.

4.4 Experiments

For our experiments, we use the CASIA Iris database from the Chinese Academy of Sciences Institute of Automation (CASIA) [108], a common benchmark for evaluating the performance of iris recognition systems. For the iris feature extraction, we use the MATLAB code from [99] to generate both the iris feature vectors and the masks.

4.4.1 Homomorphic Encryption Processing

In this subsection, we summarize the complexity and communication costs of various HE based encrypted-domain processes discussed in Section 4.1. Each iris feature vector is 9600 bit long. The similarity threshold ϵ is set to be 0.35. I select 1,948 samples from CASIA based on the following criteria: the distances are smaller than

0.35 between any two samples from the same eye, and larger than 0.40 between any two samples from different eyes. Furthermore, each eye contains at least six good samples and one sample is set aside for testing. A total of 160 individuals are included in our dataset. Our Paillier implementation is based on the Paillier Library developed by J. Bethencourt [109]. The key length of the Paillier cipher is set to be 1024 bit which results in 2048-bit ciphertexts.

The communication cost is measured based on total amount of information exchanged between Bob and Alice without any overhead from the network stack. The computation time excludes networking time and is computed based on averaging 100 trials. All of them are implemented in C language on a Linux machine with a 2.4 GHz AMD Athlon 64 CPU and 2 GB memory. Table 4.1 summarizes the results. Encrypted-domain addition and multiplication with plaintext are relatively lightweight, except when the plaintext multiplier is negative (i.e. a large positive number in modular arithmetic). Multiplication between two encrypted numbers (MULT) takes the longest and requires information exchange between Bob and Alice. Hamming distance (DIST) is fast as there are no encryption or decryption. Bit extraction (EXTRACT) takes longer and threshold comparison (COMPARE) takes the longest due to the repeated use of negative numbers, encryption and decryption processes. The long computation time for Query preparation is primarily due the high dimension of the iris feature. The overall computation of an ABAC system consists of a fixed setup time of query preparation followed by the time taken for the remaining steps scaled by the size of the database. For a database of 10,000 iris, my ABAC system is estimated to take 41,490 seconds or 11.5 hours and 120 MBytes of network

Table 4.1: Time and Communication Complexities of HE based Encrypted-domain processing

Process	Bob's Time in sec.	Alice's Time in sec.	Communication (Kbits)
Encryption $Enc_{pk}(x)$	17.3×10^{-3}	-	-
Decryption $Dec_{sk}(c)$	12.8×10^{-3}	-	-
Addition $Enc_{pk}(x) \cdot Enc_{pk}(y)$	13×10^{-6}	-	-
Multiplication $Enc_{pk}(x)^y,\ y \geq 0$	0.143×10^{-3}	-	-
Multiplication $Enc_{pk}(x)^y,\ y < 0$	30.1×10^{-3}	-	-
MULT	47.9×10^{-3}	43.0×10^{-3}	3
DIST[a]	98×10^{-3}	-	-
EXTRACT[b]	0.845	0.421	56
COMPARE[b]	2.06	0.602	42
Query Preparation (Step 2 in ABAC)	-	290	-
Remaining steps in ABAC[a]	3.05	1.07	98

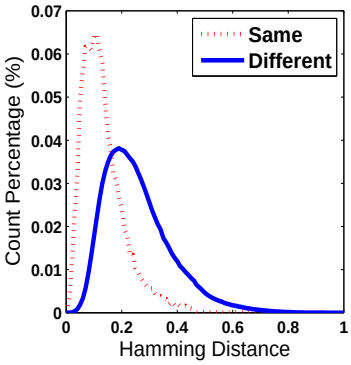
[a] Average running time for each entry in DB amortized over 100 entries, with the dimension of each entry equal to 9600.
[b] 14 bits operand are used as they are sufficient for the Hamming distance.

bandwidth.

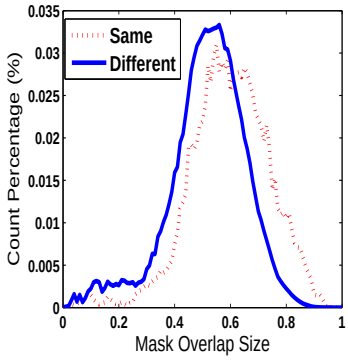
4.4.2 Privacy and Similarity among Iris Masks

In this section, I demonstrate the experiment result on similarity among iris masks which is described in Section 4.2.2. Based on the normalized hamming distance (HD), I extract $28,006$ pairs of masks between the same individuals and $7,050,197$ pairs between different individuals among 3763 samples from 292 individuals. Figure 5.2(a) shows the distribution of these two types of HDs. It can be easily found the distinct difference between the two distributions.

To further test if the difference between hamming distances from the same and different individuals are statistically significant, I utilize the distribution-free Wilcoxon Rank-Sum Test between these two samples [110, Ch.15]. I take a hypothesis test that masks from the same individual are similar to those from different individuals. If this hypothesis is accepted, there is no identity information leaked through masks and thus can be released to public; or else, the masks are one part of each individual's



(a) Hamming Distances



(b) Masks Overlap Sizes

Figure 4.3: Mask distance distributions

privacy information and cannot be shared with others.

In my test, the sample from the same indivuduals' HDs are labeled as X and the sample from different indivuduals' HDs as Y. Let u_1 and u_2 be the averages of X and Y respectively. The null hypotheses is $H_0 : u_1 - u_2 = 0$ and the alternative hypothesis is $H_a : u_1 - u_2 \neq 0$. When the samples from X and from Y are pooled into a combined sample of size $m + n$, these observations are sorted from smallest (rank 1) to largest (rank $m + n$). Then the sum of ranks of all samples from X is considered as our test statistic W, i.e. $W = \sum_{i=1}^{m} R_i$ where R_i is the rank for the i-th sample of X. Due to the large sample size, the distribution of the test statistic $z = (W - \mu_W)/\sigma_W$ can be approximated by a standard normal distribution if H_0 is true where

$$\begin{aligned} \mu_W &= \frac{m(m+n+1)}{2} = 9.91 \times 10^{10} \\ \sigma_W^2 &= \frac{mn(m+n+1)}{12} = 1.16 \times 10^{17} \end{aligned}$$

At the confident level of 99%, H_0 is rejected if either $z \geq 2.58$ or $z \leq -2.58$. In our experiments, $W = 5.19 \times 10^8$ which implies that $z = -288.91$. The null hypothesis

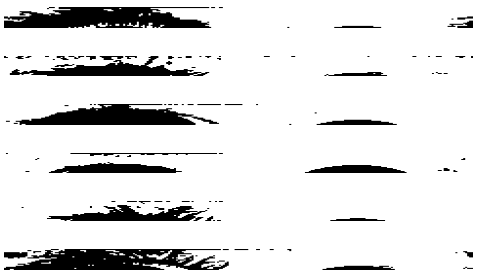
is therefore rejected.

Another illustration to demonstrate the difference between masks from the same and different individual, – the distribution of mask overlap sizes, $\|mask_{\mathbf{x}} \cap mask_{\mathbf{y}}\|$, is shown in Figure 5.2(b). It shows that masks from the same individuals have larger overlap than from different individuals. This result can also be verified by Wilcoxon Rank-Sum Test, which is omitted here as it is essentially the same as the test of the HDs. Based on these two tests, I conclude that masks have inter-correlation among each individual, and therefore, should not be shared between Alice and Bob.

4.4.3 Common Mask

Samples of masks from different individuals are shown in Figure 4.4(a). It can be observed that there are a great deal of similarity among masks even from different individuals. Also, my earlier experiments depicted in Figure 5.2(a) indicate that there could be up to 50% bit difference even between masks from the same individual. As such, it is conceivable to use a common mask to replace individual masks without much loss in precision. As I have pointed out earlier, the use of a common mask can significantly reduce the complexity of our GC circuits. To test my hypothesis, I use the following method to derive the common mask: first, I pre-align all iriscodes in Bob's database, both features and masks, to the position which can get the minimum Hamming distance when comparing with one randomly chosen iriscode from the same individual. The common iris mask is set to '1' at all bit positions where the percentages of the pre-aligned masks being '1' at those positions exceed an empirically-determined threshold λ. The common mask obtained from the CASIA

iris database is shown in Figure 4.4(b).



(a) Real masks from database

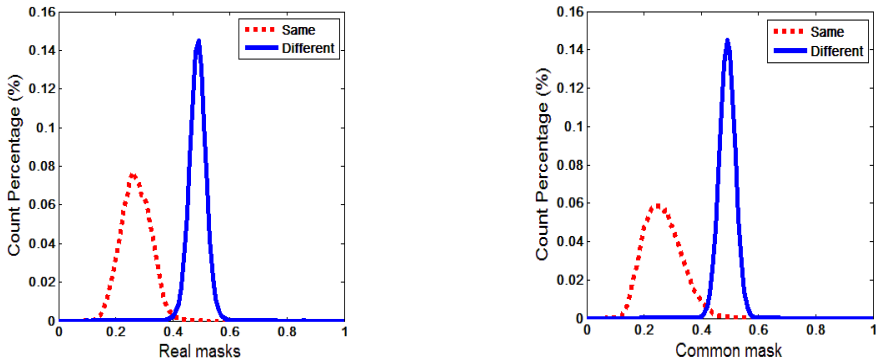
(b) Common mask

Figure 4.4: Real masks and common mask

Figure 4.5 shows the distribution of HDs using both real masks and the common mask. When $\epsilon = 0.41$, False Accept Rate (FAR) is 0.53% while False Reject Rate is 0.54% for the distribution computed with real masks. The best FAR and FRR is 1.44% and 1.47% at $\epsilon = 0.43$ for the distribution with the common mask, based on setting λ to 80%. It can be seen that the accuracy in the case of common mask is reduced by less than 1%.

4.4.4 Garbled Circuits Processing

I analyze the results using two sets of iriscodes – the length of an iriscode is $n = 2048$-bit based on the system by Daugman [27] and $n = 9600$-bit based on an open source iris recognition system in [99]. Since we do not have the original 2048-bit iriscode, I generate it by subsampling the 9600-bit iriscode. The use of the generated 2048-



(a) Real masks (b) Common mask

Figure 4.5: HD distributions

bit iriscode does not affect the computation and communication complexity of the proposed framework.

I do not analyze the precomputation for circuit construction and circuit transmission since they are executed only once. I count the precomputation for oblivious transfer as the offline time since it needs to be done every time when our protocol is implemented [53]. The offline time is independent of the size of biometric database but related to the length of the iriscode, as shown in Table 4.2. Table 4.2 also lists the total amount of non-XOR gates and runtime needed to implement the sub-circuit to test if $D(q, X_i) < \epsilon \cdot M(q, X_i)$, together to the total amount of data transmitted during the online computation. The results are derived by averaging the comparisons of 100 pairs of iriscodes in the database.

The performance of the totally GC-based private iris-code matching is quite efficient: when I adopt 80-bit security parameter, it takes 563 ms to compare two 2048-bit iris-codes with private iris features and masks. If the common mask is used, a speedup factor of up to 8.7 or 65 ms per comparison can be achieved. This is

comparable to 14 ms as reported in [31] but with a pure GC implementation.

Considering that longer cyphertexts will be required to guarantee security with the rapid development in computational capability, I also list the processing time with the longer term security parameters (112 and 128 bits) in Table 4.2. The execution time is increased by 11% for the individual masks and 23% for the common mask. These are much smaller than the 62% increase for the hybrid protocol as reported in [32]. As such, our GC-only protocol is clearly preferred in the cases when longer term security is needed.

Table 4.2: Number of non-XOR gates, runtime (ms) and bandwidth (KB) based on different secure parameters (bit)

n-bit	# non -XOR	Sec Para.	Offline Time	Online Time		Overall Time	Band-width
				Alice	Bob		
Individual Masks							
2048	8349	80	19,767	40	108	563	571.5
		112	20,260	49	113	606	754.0
		128	20,425	61	109	608	845.7
9600	38654	80	90,744	102	508	2530	2655.0
		112	93,441	106	539	2769	3503.2
		128	92,736	128	557	2816	3828.5
Common Mask							
2048	2059	80	10,379	11	24	65	133.7
		112	10,396	11	29	74	176.5
		128	10,399	16	30	80	197.9
9600	9641	80	45,354	26	115	538	626.1
		112	45,431	28	119	545	826.5
		128	45,313	57	130	573	926.7

4.4.5 Comparison of Complexity and Communication Costs on HE, GC, and SSS

In this section, I discuss the complexity and communication costs of various private processes using three different cryptographic schemes, –HE, GC, and SSS. My

Table 4.3: Comparison among HE, GC, and SSS (Computation (Comp.) time: us; Communication (Comm.) bandwidth: byte)

security parameters	SS		HE		GC	
	infinitely		1024 bits		80 bits	
Process	Comp.	Comm.	Comp.	Comm.	Comp.	Comm.
Encryption	0.39	24	17,474.38	256	825.35	161
Decryption	0.12	24	33,518.59	257	21.89	81
Addition	0.12	-	30.60	-	2,025.05	320
Multiplication	0.14	-	219.79	-	22,729.64	4800
Renormalization	0.63	48	-	-	-	-
Compare	27.81	600	2.662×10^6	5.25×10^3	2,235.15	320

experiments are implemented in Java language on an Intel Core2 Duo CPU E8400 @3.00GHz 3.00GHz with 8GB RAM on 64-bit windows 7 Professional. In Table 4.3, all operands are 8-bit positive numbers. Three non-collude parties take part in the computation of SSS and all secret shares are 64-bit long integer, while HE adopts 1024-bit security parameter and correspondingly, GC uses 80-bit security parameter. Both HE and GC can only guarantee short-term security under such security parameters, which means that maximum expected security life of encrypted data is five years [91]. Even on such short security guarantee compared to SSS's infinity security, HE and GC are still inferior to SSS no matter from the aspect of computation or communication complexity. Therefore, when the third party can be found to assist the computation of ABAC, it is a better way to use SSS as a secure primitive to protect the privacy information of both Alice and Bob from each other.

Chapter 5

Privacy-complexity Tradeoff in CS-SMC

In Chapter 4, I show that both the complexities and the communication costs of ABAC depend linearly on the size of the database, making anonymous subject identification difficult to scale to large databases. Inspired by the k-anonymity model, a simple approach is to tradeoff complexity with privacy by quickly narrowing Alice's query into a small group of k candidates and then performing the full cryptographic search only on this small group. k will serve as a parameter to balance between the complexity and the privacy needed by Alice. This is the idea behind the k-Anonymous Quantization (kAQ) which is the main topic of this chapter.

5.1 k-Anonymous Quantization (kAQ)

The functional definition of an Anonymous BAC (ABAC) system has been given in Section 2.1.2. In this section, parameter k is added into the definition of ABAC and limit the execution of ABAC in a k-member group. The definition of *k-Anonymous BAC* (k-ABAC) procedure is as follows:

Definition 3 *An* k-Anonymous BAC *(k-ABAC) procedure is a BAC system on Bob's database DB and Alice's probe* $\mathbf{q}$ *with the following properties at the end of the protocol:*

1. *There exists a subset* $S \subset DB$ *with* $|S| \geq k$ *such that for all* $\mathbf{x} \in DB \setminus S$*, Bob knows* $d(\mathbf{q}, \mathbf{x})^2 \geq \epsilon$*.*

2. *Except for the value* y_{BAC} *as defined in Definition 1, Bob has negligible knowledge about* $\mathbf{q}$ *and* $d(\mathbf{q}, \mathbf{x})$*, for all* $\mathbf{x} \in DB$*, as well as the comparison results between* $d(\mathbf{q}, \mathbf{x})^2$ *and* ϵ *for all* $\mathbf{x} \in S$*.*

3. *Except for the value* y_{BAC}*, Alice has negligible knowledge about* ϵ*,* $\mathbf{x}$*,* $d(\mathbf{q}, \mathbf{x})$*, and the comparison results between* $d(\mathbf{q}, \mathbf{x})^2$ *and* ϵ *for all* $\mathbf{x} \in DB$*.*

The definition of k-ABAC system is similar to that of ABAC except that Bob can prematurely exclude $DB \setminus S$ from the comparison. Even though Alice may be aware of such a narrowing process, the k-ABAC has the same restriction on Alice's knowledge about DB as the regular ABAC. There are two challenges in designing a k-ABAC system:

1. How do we find S so that the process will disclose as little information as possible about $\mathbf{q}$ to Bob?

2. How can Alice choose S that contains the element that is close to $\mathbf{q}$ without learning anything about DB?

The following sections describe my approaches to solve these problems in the context of iris matching.

5.1.1 Basic Formulation and Assumptions

A direct consequence of Definition 3 is that if there exists a $\mathbf{x} \in DB$ such that $d(\mathbf{q}, \mathbf{x})^2 < \epsilon$, $\mathbf{x}$ must be in S. In order to achieve the goal of complexity reduction, our approach is to devise a static quantization scheme of the feature space F^n and publish it in a scrambled form so that Alice can select the right group on her own. To explain this scheme, let us start with the definition of a ϵ-ball k-quantization. Define $B_\epsilon(\mathbf{x})$ or the ϵ-ball of $\mathbf{x}$ to be the smallest subset of F^n that contains all $\mathbf{y} \in F^n$ with $d(\mathbf{y}, \mathbf{x})^2 < \epsilon$. An ϵ-ball k-quantization of DB is defined below:

Definition 4 *An* ϵ-ball k-quantization *(eBkQ) of* DB *is a partition* $\Gamma = \{P_1, \ldots, P_N\}$ *of* F^n *with the following properties:*

1. $\bigcup_{i=1}^{N} P_i = F^n$ *and* $P_i \cap P_j = \phi$ *for* $i \neq j$.

2. *For all* $\mathbf{x} \in DB$, $B_\epsilon(\mathbf{x}) \cap P_j = B_\epsilon(\mathbf{x})$ *or* ϕ *for* $j = 1, \ldots, N$.

3. $|DB \cap P_j| \geq k$ *for* $j = 1, \ldots, N$.

Property 1 of Definition 4 ensures that Γ is a partition while property 2 ensures that no ϵ-ball centered at a data point straddles two cells. The last property ensures that each cell must at least contain k elements from DB. The importance of using an eBkQ Γ is that if Γ is a shared knowledge between Alice and Bob, Alice can select $P_j \ni \mathbf{q}$ and communicate the *cell index* j to Bob. Then Bob can compute $S := DB \cap P_j$ which must contain, if exists, any $\mathbf{x}$ where $d(\mathbf{q}, \mathbf{x})^2 < \epsilon$.

While a typical vector quantization of DB will satisfy the ϵ-ball preserving criteria, the requirement of preserving the anonymity of $\mathbf{q}$ imposes a very different constraint.

The cell P_j that contains Alice's probe $\mathbf{q}$ should reveal as little information about $\mathbf{q}$ as possible. Individuals in the same cell may come from the same family, have the same skin color or from the same ethnic group – any common traits among the database elements within P_j can be taken advantage of by Bob to find out more about Alice, effectively lowering the privacy parameter k. To maintain k as high as possible, it is reasonable to make all elements with each cell as *dissimilar* as possible. Collectively, we want to design an eBkQ such that even the "smallest" cell is maximally dissimilar. This leads to our definition of k-Anonymous Quantization (kAQ):

Definition 5 *An* optimal k-anonymous quantization Γ^* *is an eBkQ of DB that maximizes the following utility function among all possible eBkQ* Γ*:*

$$\min_{P\in\Gamma} \sum_{\mathbf{x},\mathbf{y}\in P\cap DB} d(\mathbf{x},\mathbf{y})^2 \tag{5.1}$$

The utility function (5.1) can be interpreted as the total dissimilarity of the most homogeneous cell P in the partition. The utility function also depends on the number of data points in a cell – adding a new point to an existing cell will always increase its utility. Thus finding the partition that maximizes this utility function not only can ensure the minimal amount of dissimilarity within a cell, it also promotes equal distribution of data points among different cells.

A key assumption behind (5.1) is that the closeness between two biometric signals $\mathbf{x}$ and $\mathbf{y}$ may indicates certain relationship between the two associated individuals. This argument is certainly conceivable for biometric signals such as face images as family members certainly share similar facial features [111]. Prior experimental results show that fingerprints and palmprints from twins have some inherent correlation and

are more similar to each other than those from random individuals [112, 113]. It is less clear whether the same will apply for highly discriminative iris patterns. In [11], we have demonstrated that twins irises are more similar compared with random pairs of unrelated individuals. The details of our experimental results on comparing the modified hamming distances among twins and non-twins iris patterns are provided in Section 5.2.2.

It is important to realize that these experiments do not refute the validity in using iris patterns to distinguish twins. Indeed, the variability among irises from the same individual are significantly smaller than those between twins and thereby support the procedure of using a similarity distance threshold for human identification. As demonstrated by the data in Section 5.2.2, there is significant overlap in distances between the populations of twins and non-twins irises and it will be unreliable to separate the two populations with a single threshold. In kAQ, we maximize the utility function defined in (5.1) as a way to exclude any possibility of grouping twins in the same cell. Such an one-sided application is certainly justified by the experimental results in Section 5.2.2.

5.1.2 Neighborhoods

It is challenging to solve for the optimal kAQ for the iris matching problem due to the high dimension, 9600 to be exact, and the uncommon distance used. Our approach is to project this high dimensional space into a lower dimensional Euclidean space $\Re^m$ by using Fastmap followed by PCA. The Fastmap is used to embed the native geometry of the feature space into an Euclidean space while the PCA optimally minimizes the

dimension of the resulting space. Even in this lower dimensional space, the structure of a quantization, namely the boundary of individual cells, can still be difficult to specify. To approximate the boundary with a compact representation, we first use a simple uniform lattice quantization to partition $\Re^m$ into a rectilinear grid Ω consisting of L bins $\{B_1, \ldots, B_L\}$. Then a second-level structure, which we call *neighborhood*, is needed to group together all the bins that can possibly contain patterns from the same individual. In this section, we describe the design of neighborhood structure. In Section 5.1.3, we describe a greedy algorithm that Bob runs to group neighborhoods into an optimal kAQ structure. Finally in Section 5.1.4, we describe the encrypted-domain processing required in jointly computing the dimension reduction procedure and selecting the appropriate cell for a given probe.

Assuming that multiple training patterns are available for each individual, it is natural to estimate the neighborhood structure using the training data. There are three fundamental requirements of the design of the neighborhood structure pertinent to the overall kAQ scheme:

1. **Recognition:** The neighborhood structure must provide a high recognition rate, i.e. the error probability that any iris pattern from an individual falls outside the trained neighborhood of the same individual must be negligible. This guarantees that individuals in the biometric database will be appropriately protected.

2. **Overlap among neighborhoods:** The second type of error, where a pattern falls into someone else's neighborhood, does not affect recognition – the sub-

sequent step of encrypted-domain processing will compute the actual distance and realize that this is not a match. On the other hand, if this pattern indeed corresponds to an individual in the database, this implies that there may be an overlap between neighborhoods, or at least the coarse approximation of the neighborhoods using the bins as illustrated in Figure 5.1. If these two neighborhoods belong to two different cells as in Figure 5.1, the complexity of the subsequent encrypted-domain processing essentially doubles. As such, it is imperative to minimize the amount of overlap among neighborhoods.

3. **Ease of Computation:** There are two aspects to this requirement. First, it is beneficial to have a simple computational procedure, such as a bounding box or an ϵ-ball, to characterize a neighborhood so as to facilitate the algorithm in determining the cell membership of each neighborhood. Second, as mentioned in Section 1.4, the kAQ will produce a public table that maps bin indices in each neighborhood to the corresponding cell ID. The size of this table strongly depends on the neighborhood structure as we shall explain later.

The two most intuitive neighborhoods are bounding boxes and ϵ-balls. An ϵ-ball of an individual contains all the bins whose centroids are within ϵ from the centroid of the training patterns. A ball is simply defined to be the smallest ϵ-ball that contains all the training patterns. For our target database of iris biometric patterns, we have experimented with four different neighborhood structures:

1. ϵ-ball with a constant ϵ equal to the maximum radius of all the balls;

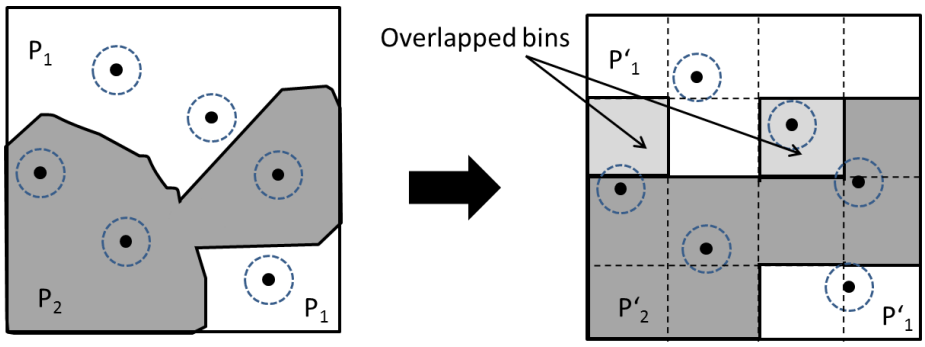


Figure 5.1: *Approximation of the quantization boundary (left) along the bins (right): each dot represent a biometric signal with the broken-line circle representing the neighborhood. There are two cells and each cell has $k = 3$ neighborhoods. As the bins do not exactly coincide with the neighborhoods, there are two neighborhoods that straddle on both cells. These "overlapping" neighborhoods cause the two cells need to be merged together, thereby raising the complexity of subsequent steps.*

2. ϵ-ball with a constant ϵ equal to the average radius of all balls plus one standard deviation;

3. ϵ-ball with the actual radius of each ball, and

4. bounding box.

In Section 5.2.3, we experimentally demonstrate that ϵ-balls provide better performance than bounding boxes in terms of the above criteria.

5.1.3 Greedy kAQ

We maximize the utility function (5.1) but require each cell to be composed of neighborhoods, each of which contains multiple bins. This turns an optimal partitioning problem in continuous space into a discrete knapsack problem in assigning bins to cells through a mapping function f to optimize the utility function. We denote the resulting approximated k-quantization as $\widehat{\Gamma^*}$. As the utility function (5.1) is based

on individual data points, a bin straddling multiple neighborhoods may present in multiple cells. As such, $\widehat{\Gamma^*}$ is no longer a true partition and the mapping function f is a multi-valued function. Protocol 12 (KAQ) describes a greedy algorithm that computes a sub-optimized k-anonymous quantization mapping function from the data.

Step 1 of KAQ sets the number of cells to be the maximum and the protocol will gradually decrease it until each cell has more than k data points. The initialization steps in 2 and 3 randomly assign a neighborhood into each cell. Step 4 identifies the cells that have the minimum utility. Among these cells, steps 5 and 6 identify the cell P_{i^*} and the neighborhood NS^* which together produce the maximum gain in utility. The bins inside NS^* are then added to P_{i^*} and the whole process repeats. This update not only provides a greedy maximization of the overall utility function but also has the tendency to produce an even distribution of data points among different cells. A newly updated cell will have a much lower chance of being updated again as it has a higher utility than others. The final step checks to see if any one cell has less than k elements and, if yes, restarts the process with fewer target number of cells. For a fixed target number of cells, the complexity of this greedy algorithm is $O(M^2)$ where M is the size of DB. It is important to point out that the output mapping f only contains entries of bins that belong to at least one neighborhood.

5.1.4 Secure Index Selection

Let us first describe how Alice and Bob can jointly compute the projection of Alice's probe $\mathbf{q}$ into the lower dimensional space formed by Fastmap and PCA. The projection

Protocol 12 Greedy k-Anonymous Quantization KAQ

Require Bob: Projection of DB into $\Re^m$ or $\{P(\mathbf{x}_i)$ for $i = 1, \ldots, M\}$; Bin and neighborhood structures in Ω;

Ensure Bob computes the multi-valued mapping $f : \Omega \to \{1, \ldots, N\}$ that defines the cell membership of each bin.

1. Set the initial number of cells $N := \lfloor M/k \rfloor$.
2. Let $L :=$ the list of neighborhoods in Ω
3. Random initialization of cells: for $i = 1, \ldots, N$,
   a) Randomly remove a neighborhood NS from L.
   b) Set $f^{-1}(i) := \{$bins in $NS\}$.
4. Identify the collection of cells E with the lowest utility, i.e.
$$E := \arg\min_{i=1,\ldots,N} \sum_{\mathbf{x},\mathbf{y} \in A_i \cap DB} d(\mathbf{x}, \mathbf{y})^2$$
   where $A_i = \bigcup_{B \in f^{-1}(i)} B$ contains all the bins in cell i.
5. For each cell j in E, identify the neighborhood $NS_j^* \in L$ that maximizes the utility of cell j after adding NS_j^* to it and denote the resulting utility as u_j^*, i.e.
$$NS_j^* := \arg\max_{NS \in L} \sum_{\mathbf{x},\mathbf{y} \in (A_j \cup NS) \cap DB} d(\mathbf{x}, \mathbf{y})^2 \tag{5.2}$$
$$u_j^* := \sum_{\mathbf{x},\mathbf{y} \in (A_j \cup NS_j^*) \cap DB} d(\mathbf{x}, \mathbf{y})^2 \tag{5.3}$$
6. Given $j^* = \arg\max_{j \in E} u_j^*$, identify the neighborhood $NS^* := NS_{j^*}^*$ and cell P_{j^*} that give rise to the maximum gain of utility from step 5.
7. Set $f^{-1}(j^*) := f^{-1}(j^*) \cup \{$bins in $NS^*\}$ and remove NS^* from L.
8. Go back to Step 4 until L is empty.
9. For $i = 1, \ldots, N$, ensure that $\left| \bigcup_{B \in f^{-1}(i)} B \cap DB \right| \geq k$. If not, set $N := N - 1$ and go back to step 2.

needs to be performed in encrypted domain so that Alice does not reveal anything about her probe and Bob does not reveal any information about his database, the Fastmap pivot points and the PCA basis vectors. Note that the need for encrypted-domain processing does not affect the scalability of our system as the computation complexity depends only on the dimension of the feature space but not on the size of the database.

The Fastmap projection in Equation (2.3) involves a floating point division. The typical approach of pre-multiplying both sides by the divisor to ensure integer-domain computation does not work. As the Fastmap update Equation (2.4) needs to square the projection, recursive computation into higher dimensions will lead to a blowup in the dynamic range. To ensure all the computations are performed within in a fixed dynamic range, Alice and Bob need to agree on a pre-defined scaling factor α and rounding will be performed at each iteration of the Fastmap calculation. Specifically, given the encrypted probe $Enc_{pk}(\mathbf{q})$, Bob approximates the first projection q' in encrypted domain based on the following formula derived from Equation (2.3):

$$\alpha\widetilde{q'} := \text{round}\left(\frac{\alpha}{2ad}\right)\widehat{d_H}(\mathbf{q},\mathbf{x}_A)^2 + \text{round}\left(\frac{\alpha}{2cd}\right)\widehat{d_H}(\mathbf{x}_A,\mathbf{x}_B)^2 - \text{round}\left(\frac{\alpha}{2bd}\right)\widehat{d_H}(\mathbf{q},\mathbf{x}_B)^2 \tag{5.4}$$

where $a = \|mask_{\mathbf{q}} \cap mask_{\mathbf{x}_A}\|_2^2$, $b = \|mask_{\mathbf{q}} \cap mask_{\mathbf{x}_B}\|_2^2$, $c = \|mask_{\mathbf{x}_A} \cap mask_{\mathbf{x}_B}\|_2^2$ and $d = d_H(\mathbf{x}_A,\mathbf{x}_B)$. All the multipliers on the right hand side of (5.4) are known to Bob in plaintext and the distances can be computed in the encrypted domain using Procedure 2. Since rounding is involved, $\widetilde{q'}$ is just an approximation of q' as computed with in the original Fastmap formula (2.3). Based on the computed encrypted values

of $\alpha q'$ from the probe and $\alpha x'$ from a data point, the update equation (2.4) is executed as follows:

$$\alpha^2 \widetilde{d'_H}(\mathbf{x},\mathbf{q})^2 := \text{round}\left(\frac{\alpha^2}{\|mask_{\mathbf{x}} \cap mask_{\mathbf{q}}\|_2^2}\right) \widehat{d_H}(\mathbf{x},\mathbf{q})^2 - (\alpha\widetilde{x'} - \alpha\widetilde{q'})^2 \qquad (5.5)$$

Bob again can compute the right hand side of (5.5) entirely in encryption domain, with the square in the second term computed using Procedure 1. The value $\widetilde{d'_H}(\mathbf{x},\mathbf{q})^2$ is again approximated due to the rounding of the coefficient. Note that the left hand side has an extra factor of α which needs to be removed so as to prevent a blowup in the dynamic range. To accomplish that, Bob computes $Enc_{pk}(\alpha^2\widetilde{d'_H}(\mathbf{x},\mathbf{q})^2 + r\alpha)$ where r is a random number, and sends the result to Alice. Alice decrypts it, divides it by α and round it to obtain round $\left(\alpha\widetilde{d'_H}(\mathbf{x},\mathbf{q})^2\right) + r$. Alice encrypts the result and sends it back to Bob who will then removes the random number r.

Bob can now use the new distances to project the probe along the second pair of pivot objects $\mathbf{x}_{A'}$ and $\mathbf{y}_{A'}$ as follows:

$$\alpha^2\widetilde{q"} := \text{round}\left(\frac{\alpha}{2d'}\right)\alpha\widetilde{d'_H}(\mathbf{q},\mathbf{x}_{A'})^2 + \text{round}\left(\frac{\alpha^2}{2}\right) - \text{round}\left(\frac{\alpha}{2d'}\right)\alpha\widetilde{d'_H}(\mathbf{q},\mathbf{x}_{B'})^2 \quad (5.6)$$

where $d' = \widetilde{d'_H}(\mathbf{x}_{A'},\mathbf{x}_{B'})^2$ can be computed by Bob in plaintext. The extra factor of α on the left hand side of (5.6) can be removed with the help of Alice using a similar approach as previously discussed. As the iteration continues, the deviation of the rounded projection and the original projection will grow as the rounding error accumulates. However, the new distance computed at each iteration absorbs the rounding error from the previous projection. As a result, the distance in the projected space will approach the underlying distance in a similar manner as the original projection.

In the computation of PCA projection, we scale each basis vector with a large enough multiplier not only to absorb the fractional parts of the basis vector but also the scalar α used in Fastmap. Let the i^{th} basis vector of PCA be $\mathbf{p}_i = \eta(p_1^i, p_2^i, \ldots, p_{m_1}^i)^T$ where $i = 1, \ldots, m_2$ with m_2 being the target PCA dimension. The encrypted-domain PCA projection of the Fastmap projection of $\mathbf{q}$ can be computed as follows:

$$
\begin{aligned}
Enc_{pk}\left[P_{pca}(P_{fm}(\mathbf{q}))_i\right] &:= Enc_{pk}\left[P_{fm}(\mathbf{q})^T\mathbf{p}_i\right] = Enc_{pk}\left[\sum_{j=1}^{m_1}\alpha P_{fm}(\mathbf{q})_j\frac{\eta p_j^i}{\alpha}\right] \\
&= \prod_{j=1}^{m_1} Enc_{pk}\left[\alpha P_{fm}(\mathbf{q})_j\right]^{\frac{\eta p_j^i}{\alpha}} \qquad (5.7) \\
&\approx \prod_{j=1}^{m_1} Enc_{pk}\left[\alpha P_{fm}(\mathbf{q})_j\right]^{\text{round}\left(\frac{\eta p_j^i}{\alpha}\right)} \qquad (5.8) \\
& \qquad (5.9)
\end{aligned}
$$

The scalar η is selected so that the loss of precision due to rounding is sufficiently small.

The last step of the process is to quantize the projection $P_{pca}(P_{fm}(\mathbf{q}))$. We only consider the quantization step size in powers of two so that the quantization process can be performed in the encrypted domain: first, we use the secure bit extraction routine EXTRACT to compute the binary representation of $Enc_{pk}\left[P_{pca}(P_{fm}(\mathbf{q}))\right]$. Then, we drop the lower order bits based on the chosen step-size. The resulting bits are recombined to form the binary representation to the encrypted bin index $Enc_{pk}(B)$.

In order to obtain the cell index $f(B)$, we need an additional cryptographic tool: a homomorphic collision-resistant hash function $h_{PK_h}(\cdot)$ with the following homomor-

phic property [114, 115]:

$$h_{pk_h}(x+y) = h_{pk_h}(x) \cdot h_{pk_h}(y) \tag{5.10}$$

Our implementation is based on [114]. Bob generates both the public key pk_h and the secret key for this hash function, and shares the public key with Alice. Instead of directly publishing the mapping $f(\cdot)$ between the bin index and the corresponding cell indices, Bob publishes an obfuscated mapping $f'(\cdot)$ such that $f(B) = f'(h_{pk_h}(B))$. The hash function sufficiently scrambles all the bin indices so that the distribution of Bob's data among all the bins classified in the KAQ algorithm is disguised as random sampling in the range of the hash function. To prevent Alice from launching a dictionary attack on the table, the length of the bin index must be large enough. This can be accomplished, for example, by padding random projections of the query to make the bin index longer. The cell indices will be published without any obfuscation – little information is leaked through them as it is shared knowledge between Alice and Bob that there are roughly N/k distinct cell indices, each of them occurring around k times.

The reason we need the homomorphic property (5.10) is to help Alice in computing $h_{pk_h}(B)$. After Bob finishes the computation of $Enc_{pk}(B)$, he picks a random r, computes $h_{pk_h}(r)$ and $Enc_{pk}(B-r)$ and sends them to Alice. Alice then decrypts $Enc_{pk}(B-r)$, computes $h_{pk_h}(B-r)$ and uses the homomorphic property to compute $h_{pk_h}(B) = h_{pk_h}(B-r) \cdot h_{pk_h}(r)$. After that, Alice performs a table lookup to find $f'(h_{pk_h}(B)) = f(B)$. If there are multiple cell indices in $f(B)$, Alice should not send all of them to Bob because he may use this information to significantly reduce the

possible choices of B as overlapped bins are rare. Instead, Alice should send one cell index first. Then, she re-encrypts her probe and reruns the entire dimension reduction and index selection process as if she was a different user. The same $f(B)$ will be computed and Alice sends Bob the second index. The whole process is repeated until all the cell indices in $f(B)$ are exhausted or a match occurs.

SELECT (Protocol 13) summarizes the above process on how Bob can identify the cell to which $\mathbf{q}$ belongs. As for the security of Protocol 13, steps 1 through 4 are processing in encrypted domain and thus reveal no secrets to either parties. Steps 5 and 6 allow Bob to identify the cell indices to which $\mathbf{q}$ belongs. As we assume Bob to be semi-honest, Bob will not deviate from the protocol by adding any identifiable information to the public table $f'(\cdot)$. Alice has no incentive to deviate from this protocol as a wrong cell index will erase any chance of success in the subsequent encrypted-domain matching with the elements in the cell. The complexities of Protocol 13 are $O(m_1 m_2 + m_2 l)$ on Bob side and $O(m_2 l)$ on Alice side, where m_1 is the Fastmap dimension, m_2 is the PCA dimension and l is the bit length of the scaled PCA coordinates. The communication costs are $O(m_1 + m_2 l)$ encrypted numbers.

Protocol 13 Secure Cell Index Selection SELECT

Require Alice: Probe $\mathbf{q}$; Bob: Fastmap pivot objects, PCA basis, and quantization step-size in PCA space, $\{2^{q_i}$ for $i = 1, \ldots, m_2\}$; Public: Scrambled Mapping $\tilde{f}$, Deterministic homomorphic cipher with unknown secret key $Enc_{pk^*,r^*}(\cdot)$

Ensure Bob gets $f(B)$ where $B \in \Omega$ contains $\mathbf{q}$

1. Alice and Bob computes $Enc_{pk}\left[P_{pca}(P_{fm}(\mathbf{q}))_i\right]$ for $i = 1, \ldots, m_2$.
2. Bob creates an empty list $G := \phi$.
3. Quantization of the projection: for $i = 1, \ldots, m_2$,
   a) Bob and Alice execute $R :=$ EXTRACT$[Enc_{pk}\left(P_{pca}(P_{fm}(\mathbf{q}))_i\right)]$ to get the encrypted binary representation of the i^{th} dimension of the projection of $\mathbf{q}$.
   b) Bob discards q_i lower order encrypted bits from R and add the remaining bits to the set G.
4. Bob recombines individual encrypted bits in G to create a single encrypted $Enc_{pk}(B)$.
5. Bob generates a random number r, compute and sends Alice $Enc_{pk}(B-r)$ and $h_{pk_h}(r)$.
6. Alice decrypts $Enc_{pk}(B-r)$, computes $h_{pk_h}(B) = h_{pk_h}(B-r) \cdot h_{pk_h}(r)$ and uses it look up the cell indices $f(B) = f'(h_{pk_h}(B))$.
7. If $f(B)$ has multiple cell indices, Alice will send the first one to Bob, wait for a random amount of time, re-execute this entire procedure, and sends the second cell index. The process is repeated until all cell indices in $f(B)$ are exhausted or a match occurs.

5.2 Experiments

In this section, I first list the scalable experiment results implemented on kAQ and validate a key assumption used in the kAQ scheme that privacy is better preserved by grouping iris patterns that are far apart from each other into the same cell. Furthermore, I study the impact of different neighborhood structures carefully to improve the performance of kAQ.

Table 5.1: Time and Communication Complexities of kAQ

Process	Bob's Time in sec.	Alice's Time in sec.	Communication (Kbits)
SELECT*	2149.842	3.455	5522

* Fastmap dimension $m_1 = 100$; PCA dimension $m_2 = 20$ and $l = 64$.

5.2.1 Complexity of kAQ

I have mentioned in Section 4.4.1 that for a database of 10,000 iris, my ABAC system is estimated to take 41,490 seconds or 11.5 hours and 120 MBytes of network bandwidth. On the other hand, in a k-anonymous ABAC system, the fixed setup time are the Query Preparation and the SELECT process as shown in Table 5.1. The matching complexity depends only on k but not on the size of the database, except for the rare cases in which the probe falls into an overlapped bin. Apart from these exceptions, for the same database of 10,000 iris patterns using a k-ABAC system with $k = 50$, the time required is only 650 seconds and the bandwidth is 1.3 MBytes.

5.2.2 Privacy and Biometric Similarity

To illustrate my assumption that grouping patterns close to each other in the same kAQ cell may reveal important privacy information to the biometric server, I need to show that individuals who are blood-related tend to have biometric patterns that are closer to each other than unrelated individuals. In this section, we test this hypothesis based on CASIA-IrisV3-Twins iris database – that is, the modified Hamming distances among twins are smaller than those of non-twins.

There are 3183 iris images from 100 pairs of twins in CASIA-IrisV3-Twins iris database. We extract all twins' left iris images for comparison. The feature extraction

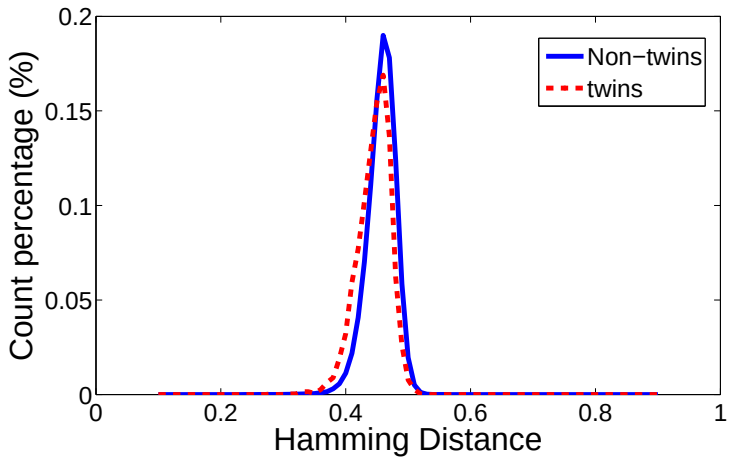


Figure 5.2: Distribution of IrisCode Hamming Distances

code obtains 1118 accurate iris codes, which result in 3351 Hamming Distances (HDs) between twins and 617631 HDs between non-twins. Figure 5.2 shows the distribution of these two types of HDs. There is a significant amount of overlap between the two distributions.

As HDs are between 0 and 1 and are clearly non-Gaussian, we perform the distribution-free Wilcoxon Rank-Sum Test between these two samples to determine if there is a statistical difference between the two distributions [110, Ch.15]. We label the samples from twins' HDs as X and the samples from non-twins' HDs as Y. Let u_1 and u_2 be the averages of X and Y respectively, and m and n be the total number of samples from X and from Y. To make the size of the two set of samples comparable, 3351 random samples are randomly selected from Y so that $m = n = 3351$. The null hypotheses is $H_0 : u_1 - u_2 = 0$ and the alternative hypothesis is $H_a : u_1 - u_2 < 0$. When we pool the samples from X and from Y into a combined sample of size $m+n$, these observations are sorted from the smallest (rank 1) to the largest (rank $m+n$).

We then consider the sum of ranks of all samples from X as our test statistic W, i.e. $W = \sum_{i=1}^{m} R_i$ where R_i is the rank for the i-th sample of X. The test procedure is one-tailed since, for small W value, H_0 would be rejected in favor of H_a. Due to the large sample size, the distribution of W can be approximated by Gaussian(μ_W, σ_W^2) if H_0 is true where

$$\mu_W = \frac{m(m+n+1)}{2} = 11.2 \times 10^6 \tag{5.11}$$

and

$$\sigma_W^2 = \frac{mn(m+n+1)}{12} = 6.27 \times 10^9 \tag{5.12}$$

Our data shows that the measured $W = 9.78 \times 10^6$. Thus, the P-value of the null hypothesis in our one-sided test can be calculated as follows:

$$\begin{aligned} \text{P-value} &= Prob(W \leq 9.78 \times 10^6) \\ &\approx \Phi\left(\frac{W - \mu_W}{\sigma_W}\right) \\ &= 6.17 \times 10^{-75} \end{aligned}$$

where $\Phi(\cdot)$ is the cumulative distribution function of a standard normal random variable. The small P-value strongly suggests the rejection of the null hypothesis and suggests the alternative hypothesis. In other words, the HDs between twins are indeed smaller than the HDs between non-twins. This demonstrates the validity of the assumption used in kAQ that grouping iris patterns closer to each other in the same cell may leak important identity information as, at the very least, twins are more likely to be grouped together.

5.2.3 Neighborhood Structures in kAQ

In this section, we evaluate the effectiveness of the four neighborhood structures described in Section 5.1.1. All the binary iris patterns are first projected to a lower dimensional Euclidean feature space via a combination of Fastmap and PCA, followed by an uniform lattice quantization. We have tested three different dimensions for the feature space: $m = 10, 20, 40$, and four different quantization levels per dimension: 2, 4 and 8 bins. While higher dimension and finer quantization provide a more accurate characterization of the neighborhoods, they also demand a larger lookup table in the cell selection phase. The experiments conducted in this section are geared to study the appropriate parameters so as to provide 100% recognition rate with minimal complexity.

We use CASIA-IrisV3-Lamp iris database in all the experiments in this section. To ensure that the raw dataset allows perfect recognition, we remove a small number of noisy samples to produce a revised dataset of 1948 samples – a total of 160 individuals are included in the dataset, and all iris patterns obtained from the same individuals are at most 0.35 hamming distance from each other and at least 0.40 to the closest pattern from a different individual. Furthermore, each individual eye contains at least six good samples. We withhold one random sample for testing and use the remaining ones for building the neighborhood structure.

For each test pattern, we measure the number of neighborhoods that contain this pattern, or the overlap number, and whether the correct neighborhood is included, or the recognition rate. The ideal overlap number is one indicating that only the correct

neighborhood is returned. A large overlap number implies that many neighborhoods will be passed to the second phase of encrypted-domain processing which results in an increase in complexity. An overlap number smaller than one indicates that the lookup fails to return any neighborhoods and results in a reduction in recognition rates, though a perfect recognition is not guaranteed by a large overlap number. The average overlap numbers and average recognition rates over all the test patterns are reported in Table 5.2.

Table 5.2: Bins' overlap and recognition rate (%) in different dimensions (m)

bin	2			4			8		
	Overlap number		Rec. rate	Overlap number		Rec. rate	Overlap number		Rec. rate
	Mean	Std	(%)	Mean	Std	(%)	Mean	Std	(%)
m = 40									
(1) ϵ-ball with a maximum radius	3.11	1.9	98.75	2.76	2.16	99.38	2.97	4.12	99.38
(2) ϵ-ball with a statistical radius	0.96	0.35	91.88	1.00	0.37	92.50	0.96	0.22	95.00
(3) ϵ-ball with different ϵ	0.81	0.5245	74.38	0.80	0.56	73.12	0.76	0.42	76.25
(4) bounding box	0.54	0.53	52.50	0.50	0.52	48.75	0.24	0.43	24.38
m = 20									
(1) ϵ-ball with a maximum radius	18.63	6.83	99.38	25.74	16.94	100	28.48	26.76	100
(2) ϵ-ball with a statistical radius	2.91	1.79	91.25	3.31	2.78	94.38	2.12	2.21	96.88
(3) ϵ-ball with different ϵ	2.27	1.43	80.00	2.36	2.00	78.13	1.64	1.38	77.50
(4) bounding box	2.08	1.29	75.63	1.38	1.16	70.00	0.56	0.58	48.75
m = 10									
(1) ϵ-ball with a maximum radius	63.98	10.25	99.38	51.48	26.82	100	76.30	36.85	100
(2) ϵ-ball with a statistical radius	18.06	5.56	92.50	18.80	12.38	93.75	9.99	10.08	96.88
(3) ϵ-ball with different ϵ	12.50	4.22	80.00	11.13	7.50	80.00	6.44	5.82	79.38
(4) bounding box	14.17	3.93	83.75	9.76	6.69	80.00	2.48	2.64	66.25

From the results in Table 5.2, we first notice that for the same level of recognition, the average overlap number decreases when the number of dimensions m increases. This can be explained by the fact that a lower-dimensional feature space introduces much distortion and cannot approximate the original hamming distance well. Since high overlap numbers will increase search complexity in the subsequent encrypted-domain processing, we will focus only on higher dimension of 20 and 40.

Second, for dimensions $m = 20$ and $m = 40$, the ϵ-ball with maximum radius produces an almost perfect recognition rate but at a cost of average overlap numbers as high as 2.76 for $m = 40$ and 25.74 for $m = 20$. Comparatively, the other three schemes all have much smaller overlap numbers. Among them, only the ϵ-ball with statistical radius achieve recognition rates over 90%. The worst is the bounding box with recognition rates between 24.38% to 75.63%. It is worth noting that both of the statistical neighborhood structures, i.e. ϵ-ball with a maximum radius and ϵ-ball with statistical radius, have better recognition rates than the individual-customized neighborhood structures including ϵ-ball with different ϵ and bounding box. Customized neighborhoods are trained using only a small number of training samples from each individual and thus are prone to poor classification results. The dominance of statistical neighborhood structures in recognition rate becomes more pronounced with the increase of the quantization levels per dimension. This can be explained by the fact that finer quantization levels result in more precise characterization of the neighborhood structures.

In order to achieve 100% recognition rate among the top performers, we increase the number of test patterns to two and take the union of all the neighborhoods

returned from both patterns. If the noise from these two patterns are independent, the recognition rate should improve at the expense of slightly higher complexity due to the increase number of neighborhood returned. To test this idea, we withhold an additional sample from each individual and redo the experiments for $m = 20$ and $m = 40$. We also adjust slightly the values of the statistical radius for the best recognition rate with the least increase of the standard deviation. The results are shown in Table 5.3.

Table 5.3: Bins' overlap and recognition rate (%) with 2 - test patterns

bin	2			4			8		
	Overlap number		Rec. rate	Overlap number		Rec. rate	Overlap number		Rec. rate
	Mean	Std	(%)	Mean	Std	(%)	Mean	Std	(%)
m = 40									
(1) ϵ-ball with a maximum radius	3.91	2.25	100	3.39	2.56	100	3.36	4.15	100
(2) ϵ-ball with a statistical radius	1.78	1.09	100	2.00	1.29	100	1.32	0.73	100
(3) ϵ-ball with different ϵ	0.88	0.49	81.25	0.89	0.47	82.50	0.86	0.35	85.62
(4) bounding box	0.76	0.48	73.75	0.71	0.47	70.62	0.42	0.49	42.50
m = 20									
(1) ϵ-ball with a maximum radius	27.70	9.67	100	34.43	18.99	100	37.46	28.35	100
(2) ϵ-ball with a statistical radius	14.15	6.08	100	17.26	11.39	100	10.55	10.99	100
(3) ϵ-ball with different ϵ	2.95	1.61	86.25	2.90	2.01	86.25	2.09	1.74	86.25
(4) bounding box	2.54	1.31	86.25	1.69	1.13	81.88	0.70	0.56	61.88

The new recognition rates among all configurations of statistical neighborhood structures are 100% but the average overlap numbers increase significantly for 20 dimension. Since all the overlap numbers are above 10, that is at least 10 fold increase in the complexity of encrypted domain processing, it is unreasonable to use any schemes with in 20 dimensions. Among the two statistical schemes in 40 dimensions, ϵ ball with statistical radius have lower average overlap numbers.

As for the number of bins, there do not seem to be much difference in terms of both average overlap numbers and recognition rates. On the other hand, the price of using high dimensions and large bin numbers is a very large quantized feature space with a great number of bins. As described in Section 5.1.1, the k-AQ requires a public table that lists out all the bin indices (scrambled) within each neighborhood and the corresponding cell ID. A direct consequence of a large space is a huge table which will take up significant storage space and increase the lookup time. A simple approach to measure the size of the table would be the average number of bins per neighborhood. Another factor to consider is the scalability of the feature space – as the dimension and the number of bins increase, the quantized feature space becomes bigger but so does each neighborhood. An important question to ask is the total number of neighborhoods that can be fit within the feature space without increasing the average overlap number. The average overlap numbers measured in Tables 5.2 and 5.3 are based on our test set of 160 neighborhoods or individuals. As such, it is impossible for us to accurately measure the average overlap number when the database is 10 or 100 times bigger. We calculate the optimistic estimate of scalability based on the ratio of volumes between the whole feature space and a neighborhood.

The results are tabulated in Table 5.4.

Table 5.4: Number of bins per neighborhood and Scalability (number of neighborhoods in space) for different structures

bin	2		4		8	
	# bins/nbd	scalability	# bins/nbd	scalability	# bins/nbd	scalability
m = 40						
(1) ϵ-ball with a maximum radius	3.78×10^{11}	2.92	1.80×10^{12}	6.73×10^{11}	1.63×10^{20}	8.14×10^{15}
(2) ϵ-ball with a statistical radius	2.13×10^{10}	51.61	1.93×10^{11}	6.26×10^{12}	1.61×10^{18}	8.26×10^{17}
m = 20						
(1) ϵ-ball with a maximum radius	9.33×10^{5}	1.12	7.44×10^{6}	1.48×10^{5}	4.75×10^{11}	2.43×10^{6}
(2) ϵ-ball with a statistical radius	1.76×10^{5}	5.95	1.20×10^{6}	9.14×10^{5}	1.15×10^{10}	1.00×10^{8}

We observe from Table 5.4 that none of configurations under 2 bins are scalable enough to hold more than 52 neighborhoods and some are as low as 1.12. As the number of bins increase, both the number of bins per neighborhood and the scalability measure increase. For 20 dimensions, the scalability measures are in the range of 10^5 to 10^8 neighborhoods in the whole space. However, configurations in 20 dimensions do not produce adequate overlap numbers and we will need to use 40 dimensions. The number of bins per neighborhood increases significantly. For 4 bins in the ϵ-ball with statistical radius, the scalability is 6.26×10^{12} which is more than adequate. The number of bins per neighborhood is 1.93×10^{11}. Each bin index can be represented by 80 bits or 10 bytes and the cell ID is in the order of the size of the database. Thus, 16 bytes would be a reasonable size for one entry in the table, which implies that it will take roughly 2TB to store all the bins for one individual.

Chapter 6

Application: Privacy-protected Video Surveillance Network

In this chapter, an application based on biometric matching is implemented in video surveillance network. Anonymous subject identification is just the beginning of privacy protection of trusted people. When these trusted individuals enter into the monitoring environment, there are subsequent work to deal with their anonymous identity to further protect their privacy.

This chapter is organized as follows: after introducing four problems in privacy-protected video surveillance system in Section 6.1 and the composition of my privacy-protected video surveillance camera network in Section 6.2, I design Privacy Information Management (PIM) system that supports anonymous authentication of privacy information retrieval using biometric signals in Section 6.3. The following experiment result will show the validity of my scenario in Section 6.4.

6.1 Problems in Privacy-protected Video Surveillance System

There are four main problems needed to be addressed in developing a privacy-protected surveillance system:

1. How to protect the sensitive visual information?

2. How to recover the protected privacy information?

3. Who should be protected?

4. Who can recover the protected privacy information?

Most of the existing literature focus on the first two problems. The first problem is on approaches to obfuscate visual information of an individual so that the true identity cannot be revealed and the overall quality of the surveillance video is preserved. Many schemes have been proposed in the literature ranging from the use of black boxes or large pixels in [4], face replacement in [116], body replacement in [38], to complete object removal in complete object removal in [117]. The second problem considers the "privacy data preservation" issue – the original data must be preserved in a secure but reversible manner as they can be used to authenticate the obfuscation process and to provide defensible evidence in legal settings. Existing approaches include scrambling [118, 119] and data hiding [8].

The remaining two problems on the "who's" in privacy surveillance systems receive far less attention in the research community. The first "who" question deals with the identification of individuals whose imageries in the surveillance video require obfuscation, which has been solved with the use of convenient and highly discriminative biometric signals like iris patterns in the previous Chapters. In this chapter, I will address the second "who" question – to provide selective and anonymous access to the preserved privacy information.

The predominant approach in the literature to the second "who" question is to set up an access policy so that different groups can access different videos – for example, in a corporation, the security camera officer may have access to video contents of

all visitors but not the employees; the chief privacy officer will have access to video contents of visitors and all employees except for the executive team but the law enforcement, with a proper order from the court, will have access to the true original footage. While such a static access policy is sufficient for small organizations, individual users can quickly lose control of their privacy information in a large organization in which the membership of different access groups are highly dynamic and typically beyond the control of individual users.

In this chapter, I advocate treating the privacy visual information of an individual in the same manner as any other privacy information such as personal financial or medical information – each access of the information must require a full consent from the corresponding user. This posts a technical challenge because the surveillance system cannot associate the imagery with the unknown identity of the individual as protected by the ABAC process. To solve this problem, we propose a novel Privacy Information Management (PIM) system that uses biometric signals in encrypting the privacy video. The biometric signal acquired during the ABAC stage will be combined with a user specified passcode in encrypting a random AES key used to encrypt the video. The passcode is used to ensure that the system cannot determine the true identity of the user through exhaustive search of the biometric database. Two SMC protocols, one for encryption and one for retrieval, are developed to ensure the privacy of both the plaintext biometric signals and privacy visual information.

6.2 Privacy-Protected Video Surveillance Network

My privacy-protected video surveillance camera network is composed of a number of intelligent camera systems. The intelligent camera system is responsible for segmenting, tracking, encrypting, and obfuscating the visual imagery corresponding to each individual based on the privacy bit from ABAC [94]. Background subtraction and shadow removal are first applied to extract foreground moving blobs from the video. Objects are tracked based on the position and size of the bounding box as well as its centroid velocity. The velocity is updated at a fixed adaption rate α using the formula below:

$$v_t = \alpha v_{t-1} + (1 - \alpha)\hat{v_t} \tag{6.1}$$

where v_{t-1} is the velocity state from the previous time and $\hat{v_t}$ is the current observed velocity. A blob association process is used to associate each observed blob to the closest track within its tracking gate. A candidate track is established for each non-associated blob and it will become a formal track after receiving observations continuously for a few frames. A track will be deleted from the tracker if no observations are associated with the track for an extended period of time. Each individual track provides a temporally-consistent labeling of each image object. Image objects corresponding to protected individuals are extracted from the video, each padded with black background to make a rectangular frame and compressed using an H.263 encoder [120]. The compressed bitstreams are encrypted along with other auxiliary information used by the privacy information management system. The encryption process is described more fully in Section 6.3.1.

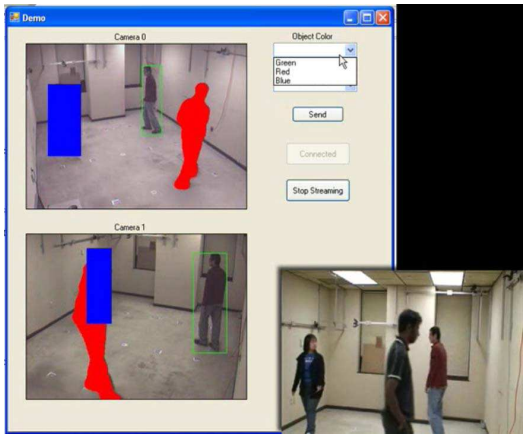


Figure 6.1: Privacy protected surveillance system

The empty regions left behind by the removal of objects can be obfuscated by a myriad of different schemes ranging from black box, colored silhouette to full object removal. Figure 6.1 shows the output of two smart cameras, alongside with a video feed at the lower right from a separate camera showing the raw video scene. Two of the three individuals are obfuscated using a blue box and a red silhouette respectively. In the lower video output, one can see that occlusion occurs between the two protected individuals. When occlusion occurs, the two objects momentarily merge with each other into a single blob and will only reappear as two blobs when the occlusion has passed. While motion segmentation during occlusion is a well-studied topic in computer vision, its accuracy is still far from perfect. To prevent privacy leakage between recorded videos for different individuals, video objects from the occluding tracks are discarded. The obfuscation will still be performed based on the segmentation scheme in [117] which uses the velocity of the bounding box and texture similarity. While Figure 6.1 shows the outputs of only two cameras, our system design is geared towards supporting a large number of cameras covering the

entire surveillance area. It is crucial to have overlapping views between cameras so as to maintain consistent object tracking from the biometric reader to all the exits. In fact, our camera placement algorithm produces a network design such that every possible object position is observed by at least two cameras [121]. The correspondences between object tracks from different camera views are established based on the "visual tagging" approach - for each blob, we use a simple ellipse fitting scheme to identify the head. The center of the head casts an epipolar line on every other camera view based on the previously estimated fundamental matrices among all pairs of cameras in the network. Intersections of epipolar lines in the proximity of a head centroid from the local camera establish the correspondences among tracks from different cameras [121]. Corresponding tracks in the two camera views in Figure 6.1 are marked with the same colored obfuscation.

6.3 Privacy Information Management (PIM)

In this section, we describe the Privacy Information Management (PIM) system that supports anonymous authentication of privacy information retrieval using biometric signals. The setting is that the server (Bob) has a database of encrypted video segments and a biometric reader at a remote computer representing a user (Alice) wants to access a privacy video segment of Alice at a given time. Alice and Bob will engage in a retrieval protocol over a public network. The security and privacy goals of this protocol, in addition to those from the ABAC module, are listed as follows:

1. Alice should have access to all of her raw video segments.

2. Alice does not have access to raw video segments of any other users.

3. Bob does not have access to any raw video segments and has no knowledge of the identity of the user associated to each video segment.

4. Bob must authenticate Alice's identity using her biometric signals but cannot have access to the raw biometric signals.

An additional design goal is to provide efficient retrieval due to the high data rate needed for processing and communicating video segments. A typical secure file system with different user accounts can easily satisfy the first two goals. However, a file system will associate each video file with an individual user, thus failing to satisfy the third goal. Using alias for a user will not work as neither the ABAC system nor the camera network provides any identity information about a video segment and thus Bob cannot tell if two different segments contains the same user. The fourth goal is different from that of the ABAC module as it is not enough to just validate Alice's membership status – the system also needs to grant access to the specific videos containing Alice's imagery. While the fourth goal requires the use of biometric signals, the fact that Bob has the biometric database should not provide him with any additional knowledge in ascertaining Alice's identity from her retrieval request.

The problems in designing the PIM module are similar to those of the Private Information Retrieval (PIR) [122], with video segments being the data records and biometric signals being the keys. Our design is in fact an adaptation of a PIR system to (1) allow variability in biometric signals between the time when the signal is first captured during ABAC and when retrieval request is made, and (2) support low

computation and communication overhead by using a surrogate data record, rather than full video in the PIR protocols. Our proposed design consists of two protocols: the first one is the encryption of the privacy imagery in video based on the biometric signal obtained during the ABAC process. The second protocol is invoked during the retrieval process where the decryption is performed using the biometric signals. These two protocols are described below.

6.3.1 Privacy Information Encryption

The privacy information encryption protocol shown in Protocol 14 is executed after Bob has ascertained, via the ABAC module, that the subject entering the surveillance needs to be protected. The biometric reader (Alice) still possesses the biometric signal in plaintext. In addition, the reader will acquire a passcode from the subject. The length of the passcode should be in the same order as the biometric signal, which means that it is likely to be generated via a pseudo-number generator based on a shorter seed sequence from the user. This passcode is not stored for identification purpose but the combination of the biometric signal and the passcode is needed for later retrieval of the privacy information.

A new party, the camera (Charlie), will join the protocol. Charlie is responsible for redacting pixels corresponding to the subject for privacy protection, encrypting the raw pixels of the subject and recording both the redacted video and privacy information to a database controlled by Bob. All three parties are assumed to be semihonest connected through a public network with plaintext biometric signals and videos treated as private information not to be shared with other parties.

Protocol 14 Privacy Information Encryption

Require *Alice*: Biometric probe $\mathbf{q}$, passcode $\mathbf{p}$ and a randomly generated keys (sk, pk) for an additive homomorphic cipher; *Bob*: Decision bit s from ABAC; *Charlie*: video segment v containing the protected subject.

Ensure Bob stores the encrypted privacy video and auxiliary information that satisfy the security goals.

1. Bob acknowledges the decision bit to Alice and Charlie. This protocol aborts if the subject requires no protection.

2. Alice sends the one-time public key pk to Charlie.

3. Charlie randomly generates an AES key k and encrypts the privacy video v to $AES(v, k)$. AES is used as it can be efficiently implemented for high-rate video data. Charlie encrypts the AES key k with pk to create $Enc(k, pk)$, and sends $AES(v, k)$ and $Enc(k, pk)$ to Bob.

4. Alice sends pk, $sk \otimes Hash(\mathbf{p})$, and $\mathbf{q} \otimes Hash(sk)$ to Bob where $\mathbf{p}$ is the passcode, $\mathbf{q}$ is the biometric probe, and $Hash()$ is a non-invertible collision-resistant hash function. The private key sk is destroyed.

5. For each video v, Bob creates a *shared video surrogate data record* R and a *private video surrogate data record* S indexed by the time of the day and the camera-ID:

$$R = \{sk \otimes Hash(\mathbf{p}), Enc(k, pk)\} \tag{6.2}$$
$$S = \{pk, \mathbf{q} \otimes Hash(sk)\} \tag{6.3}$$

   R will be shared with the requester during the retrieval process described in Section 6.3.2. The encrypted video record $AES(v, k)$ is joint indexed by R and S.

Protocol 14 does not leak any private information among different parties. The proof is as follows: Alice does not gain any information about the biometric database or videos of other users as she only receives a single decision bit from Bob about the result of her membership validation. Charlie receives pk from Alice which is randomly chosen public key and thereby gains no information about Alice. Charlie does not receive any information from Bob.

Bob receives data from both Alice and Charlie. These data include pk, $sk \otimes$

$Hash(\mathbf{p})$, $\mathbf{q} \otimes Hash(sk)$, $AES(v, k)$ and $Enc(k, pk)$. The biometric signal $\mathbf{q}$ is protected by the hash $Hash(sk)$ of the one-time secret key sk, which is protected by the hash $Hash(\mathbf{p})$ of the passcode $\mathbf{p}$. It is straightforward to see that Bob cannot gain any information about v or k from Charlie or $\mathbf{q}$, $\mathbf{p}$ and sk from Alice. The video v is protected by k which in turn is protected by pk. v can only be decrypted if and only if sk is known. sk is protected in $sk \otimes Hash(\mathbf{p})$ – while this is not a one-time pad as $Hash(\mathbf{p})$ is fixed for a given user, our protocols will ensure that sk is never available to Bob and as such Bob will never has access to $Hash(\mathbf{p})$. sk, however, will be obtained by Alice during the retrieval process as described in Section 6.3.2.

In the case when Alice's passcode $\mathbf{p}$ or $Hash(\mathbf{p})$ is stolen or eavesdropped by an attacker, $\mathbf{q} \otimes Hash(sk)$ is used to authenticate the requester as part of a GC used during the retrieval stage in Section 6.3.2. The use of GC also protects $\mathbf{q}$ against anyone who may have access to sk. As sk is random and $sk \otimes Hash(\mathbf{p})$ provides no information about $Hash(sk)$, $\mathbf{q} \otimes Hash(sk)$ provides an one-time pad encryption of $\mathbf{q}$ to prevent Bob from knowing anything about the user's identity.

6.3.2 Privacy Information Retrieval

The privacy information retrieval protocol in Protocol 15 is used when a user wants to retrieve her private video information from the video database stored at the server.

We assume that Alice represents the biometric reader which has access to both Alice's biometric signal $\mathbf{q}'$ and her passcode $\mathbf{p}'$. Note that the biometric signal $\mathbf{q}'$ can be different from that used in the encryption protocol, but their iriscode distance

Protocol 15 Privacy Information Retrieval

Require *Alice*: Probe $\mathbf{q}'$ and passcode $\mathbf{p}'$; *Bob*: Encrypted Video Database and the associated surrogate records.

Ensure Alice obtains al her private videos.

1. Alice requests Bob to send over her videos that match Alice's query on camera IDs and time of recording.
2. Bob identifies the resulting set as I_{match} and notifies the size of this set to Alice.
3. Iterate the following steps for each pair of shared and private surrogate records R_i and S_i in I_{match} for $i = 1, 2, \ldots, |I_{match}|$.
4. Bob perturbs the last field $Enc(k_i, pk_i)$ of R_i by replacing it with $Enc(k_i + r_i, pk)$ where r_i is a random number. This can be done due to the homomorphism of $Enc()$. The new shared surrogate record R_i and the corresponding encrypted video $AES(v_i, k_i)$ are sent to Alice.
5. Alice extracts $sk_i \otimes Hash(\mathbf{p_i})$ from R_i and computes $sk' \triangleq sk_i \otimes Hash(\mathbf{p_i}) \otimes Hash(\mathbf{p}')$. Alice obtains the real private key sk_i if and only if $\mathbf{p}' = \mathbf{p_i}$, provided that the hash function is collision-free.
6. $$c_i = \delta_{d(\mathbf{q}', \mathbf{q_i} \otimes Hash(sk_i) \otimes Hash(sk')) \geq \epsilon} \quad (6.4)$$
$$a_i = (c_i * s_i) \otimes r_i. \quad (6.5)$$

   The private inputs from Alice are $\mathbf{q}'$ from the reader and $Hash(sk')$ from step 5. The private inputs from Bob are $\mathbf{q_i} \otimes Hash(sk_i)$ in S_i, the threshold ϵ from the ABAC module (Section 4.2), r_i from step 4, and a new random number s_i. $d_H()$ is the modified Hamming distance using a common mask as described in Section 4.2.3. The output a_i should be un-garbled by Alice. This circuit is designed to produce $a_i = r_i$ if $sk_i = sk'$ and $d(\mathbf{q}', \mathbf{q}_i) < \epsilon$.
7. Alice decrypts $Enc(k_i + r_i, pk_i)$ from R_i by using sk' and computes $k' = k_i + r_i - a_i$, which is equal to k_i if and only if $sk' = sk_i$ (for correct decryption) and $a_i = r_i$.
8. Alice decrypts $AES(v_i, k_i)$ by using k'

under the common mask is assumed to be within the threshold ϵ. Furthermore, we assume the biometric reader in the retrieval process is *not the same reader used in the encryption process*. As such, the retrieval reader has *no access to the random*

encryption keys generated during the encryption process.

The proof of privacy of Procedure 15 is as follows. The only usable information Bob receives in this procedure is Alice's query on the target time of day and camera ID's, which are not private information. On the other hand, Alice receives a great deal of information from Bob. Bob must make sure that (i) Alice is who she claims by checking both her iriscode and passcode, and (ii) Alice can only decrypt her own videos. This can be accomplished by enforcing that Alice can only retrieve the AES keys k_i that correspond to her videos. The definition of the shared surrogate record in (6.2) implies that k_i is protected by the private key sk_i, which in turn is protected by the passcode $\mathbf{p}$ alone. This is clearly inadequate as the live probe $\mathbf{q}'$ is not used.

To incorporate $\mathbf{q}'$ in the authentication process, Bob first perturbs the encrypted AES key k_i in step 4 by adding a random r_i in the encrypted domain before sending the shared record to Alice. While step 5 allows Alice to recover the private key sk', the random r_i can only be revealed by a joint execution of the GC in step 6. Note that the condition (6.4) in step 6 may return 0 even if $sk' \neq sk_i$ but $Hash(sk') \otimes Hash(sk)$ happens to negate the differences between $\mathbf{q}'$ and $\mathbf{q_i}$. The occurrence of this situation should be extremely unlikely, and the fact that $sk' \neq sk_i$ prevents the AES k from being decrypted even if r_i is revealed. Steps 7 and 8 completes the remaining steps by recovering first the AES key and finally the actual video.

6.4 Experiments

In this section, we focus on the implementation details and performance analysis of the PIM system as described in Section 6.3. The privacy information encryption

protocol described in Protocol 14 relies on a public-key homomorphic cipher, a hash function, and an AES cipher for protecting the original video imageries. We use the Paillier cryptosystem for the public-key cipher because of its additively homomorphic property over a large dynamic range for computation [87]. Our Paillier implementation is based on the Paillier Library developed by J. Bethencourt [109]. The key length of the Paillier cipher is set to be 1024 bit which results in 2048-bit ciphertexts. The 1024-bit security parameter guarantees short-term security for up to year 2014 [91]. A large table of pre-generated private and public keys are stored at the reader which randomly selects a key pair for each entering individual. As we need a hash function that can generate a hash value with the same length as the Paillier encrypted ciphertext, we simply use a deterministic version of Paillier with a fixed key as the hash function. For video encryption, we use a 256-bit AES system and employ the video encryption model in [123, Ch.5] to encrypt the H.263 video bitstream pertinent to each individual. The encryption time of the 256-bit AES key using the Paillier system is on average 17.47 ms. The decryption time is 33.52 ms and the encrypted-domain addition amounts to 30.60 μs. The main computation and storage burden of the entire protocol are dominated by the AES encryption the private video, which varies depending on the number of protected individuals and the time duration of each protected individual inside the surveillance perimeter. Since the AES implementation is not our original work, we do not analyze the computation and communication complexity of video encryption. The privacy information retrieval protocol described in Protocol 15 needs an additional GC circuit to match the live probe from the requester with the stored data. To match the short-term secu-

rity of Paillier, we adopt 80-bit security parameter for our GC implementation. The result GC circuit has 2071 non-XOR gates with total runtime measured at 215ms for comparing a pair of 2048-bit iriscode and 9655 non-XOR gates at 549ms for 9600-bit iriscodes.

Chapter 7

Conclusions and Future directions

In this dissertation, I have proposed Anonymous Biometric Access Control (ABAC) and Privacy Information Management (PIM) that enable the anonymous use of biometric signals in a privacy-aware video surveillance system. For ABAC, my research focuses on the use of iris patterns to determine the privacy protection status of an incoming individual. While HE based ABAC is intuitive, recent advancements makes GC a more attractive choice for ABAC. In addition, one can exploit the nature of iris code in further reduce the matching complexity. I have discovered that the complexity of the GC implementation heavily depends on the use of individual iris masks. My experiments have demonstrated that while making the masks public as suggested by other work can leak privacy information, using a common mask for all comparisons can significantly reduce the complexity with negligible loss in recognition accuracy. To further reduce the computational and communication complexities, I have proposed a framework called the k-Anonymous ABAC system that tradeoffs privacy and complexity by quantizing the search space into cells, each of which contains at least k members. Complexity is reduced by restricting the encrypted domain search process to a small number of cells. Privacy is measured by the dissimilarity of the smallest cell. A greedy quantization scheme on a reduced-dimensional space called

k-Anonymous Quantization has been devised to derive the optimal quantization that maximizes privacy. Secure procedures have been proposed to perform the dimensional reduction and cell lookup.

As an application of anonymous biometric matching in privacy-preserving environment, I have designed a PIM system that protects all surveillance videos with privacy information and allows any user to anonymously access his/her own imageries. The proposed system uses the user's biometric signal and a secret passcode obtained during the ABAC process to encrypt a secret key for unlocking the original video imagery. The retrieval process is based on a combination of homomorphic encryption and GC to authenticate the identity of the user, while guaranteeing that the user cannot gain any information about other users, and the system knows nothing about the identity of the user or the actual video contents.

Future work include validation of the common mask assumption with a larger database, improved performance in SMC-based similar iris search through hierarchical clustering of data, and a distributed implementation of the PIM system in a large camera network. Also, using cloud-based distribution computing to anonymously match biometric signals, especially under malicious mode with the involvement of untrusted third computing party, could be a new direction for efficient anonymous biometric matching in privacy-preserving environment.

Bibliography

[1] L. M. Brown. Tampa drops face recognition system. Technical report, CNET, http://news.cnet.com/Tampa-drops-face-recognition-system/2100-1029_3-5066795.html?tag=nl, August 2003.

[2] J. Stanley and B. Steinhardt. Bigger Monster, Weaker Chains: The Growth of an American Surveillance Society, New York: ACLU, 2003.

[3] A. M. Berger. *US Patent 6,067,399: Privacy mode for acquisition cameras and camcorders*. Sony Corporation, May 23 2000.

[4] J. Wada, K. Kaiyama, K. Ikoma, and H. Kogane. *Monitor camera system and method of displaying picture from monitor camera thereof*. Matsushita Electric Industrial Co. Ltd., April 2001.

[5] T. E. Boult. Pico: Privacy through invertible cryptographic obscuration. In *Proc. Computer Vision for Interactive and Intelligent Environents: The Dr. Bradley D. Carter Workshop Series*. University of Kentucky, 2005.

[6] J. Wickramasuriya, M. Datt, S. Mehrotra, and N. Venkatasubramanian. Privacy protecting data collection in media spaces. In *ACM International Conference on Multimedia*, New York, NY, October 2004.

[7] J. Zhao and SC Cheung. Multi-camera surveillance with visual tagging and generic camera placement. In *Proceedings of ACM/IEEE International Conference on Distributed Smart Cameras*, 2007.

[8] J. K. Paruchuri, S.-C. S. Cheung, and M. W. Hail. Video data hiding for managing privacy information in surveillance systems. *EURASIP Journal on Information Security*, 2009(236139), 2009.

[9] J. Schiff, M. Meingast, D. Mulligan, S. Sastry, and K. Goldberg. Respectful cameras: Detecting visual markers in real-time to address privacy concerns. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 971–978. Springer, 2007.

[10] E. N. Newton, Latanya Sweeney, and B. Main. Preserving privacy by de-identifying face images. *IEEE transactions on Knowledge and Data Engineering*, 17(2):232–243, February 2005.

[11] Y. Luo, S. Ye, and S-C S. Cheung. Anonymous subject identification in privacy-aware video surveillance. In *IEEE International Conference on Multimedia & Expo (ICME2010)*, 2010.

[12] Axcess International Inc. "Active RFID/RTLS is changing the world. for the better". *http://www.axcessinc.com/solutions/solutions.html*.

[13] Juan Vargas. Scanners. *http://www.baitoaprimero.net/2012/02/scanners.html*.

[14] S.-C. S. Cheung and T. Nguyen. Secure multiparty computation between distrusted networks terminals. *EURASIP Journal on Information Security*, 2007. Article ID: 51368.

[15] Dan Boneh, Eu-Jin Goh, and Kobbi Nissim. Evaluating 2-dnf formulas on ciphertexts. In Joe Killian, editor, *Proceedings of Theory of Cryptography Conference 2005*, volume 3378 of *LNCS*, pages 325–342. Springer-Verlag, 2005.

[16] G. Aggarwal, N. Mishra, and B. Pinkas. Secure computation of the kth ranked element. In *Proceedings of Advances in Cryptology - EUROCRYPT 2004: International Conference on the Theory and Applications of Cryptographic Techniques*, pages 40–55, 2004.

[17] M. Naor and B. Pinkas. Efficient oblivious transfer protocols. In *Proceedings of SODA 2001 (SIAM Symposium on Discrete Algorithms)*, pages 448–457, Washington D.C., Jan 2001.

[18] Moni Naor and Kobbi Nissim. Communication complexity and secure function evaluation. *Electronic Colloquium on Computational Complexity (ECCC)*, 8(062), 2001.

[19] Christian Cachin, Jan Camenisch, Joe Kilian, and Joy Muller. One-round secure computation and secure autonomous mobile agents. In *Automata, Languages and Programming*, pages 512–523, 2000.

[20] M. Naor and B. Pinkas. Oblivious transfer and polynomial evaluation. In *Proc. 31st Annual ACM Symposium on Theory of Computing*, pages 554–567, 1999.

[21] M. O. Rabin. How to exchange secrets by oblivious transfer. Technical Report TR-81, Harvar Aiken Computation Laboratory, 1981.

[22] Aviel D Rubin. Security considerations for remote electronic voting. *Communications of the ACM*, 45(12):39–44, 2002.

[23] Miao Pan, Xiaoyan Zhu, and Yuguang Fang. Using homomorphic encryption to secure the combinatorial spectrum auction without the trustworthy auctioneer. *Wireless Networks*, 18(2):113–128, 2012.

[24] Jin Li, Qian Wang, Cong Wang, Ning Cao, Kui Ren, and Wenjing Lou. Fuzzy keyword search over encrypted data in cloud computing. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–5. IEEE, 2010.

[25] Karim El Defrawy and Gene Tsudik. Alarm: anonymous location-aided routing in suspicious manets. *Mobile Computing, IEEE Transactions on*, 10(9):1345–1358, 2011.

[26] Y. Luo, S-C S. Cheung, and S. Ye. Anonymous biometric access control based on homomorphic encryption. In *IEEE International Conference on Multimedia & Expo*, Cancun, Mexico, June 2009.

[27] John Daugman. How iris recognitionworks. *IEEE Transactions on Circuits and Systems for Video Technology*, 4:21–30, Jan. 2004.

[28] Y. Luo, S.-C. Cheung, T. Pignata, R. Lazzeretti, and M. Barni. An efficient protocol for private iris-code matching using garbled circuits. In *IEEE International Conference on Image Processing (ICIP 2012)*, Orlando, Florida, USA, 2012.

[29] Sayed M SaghaianNejadEsfahani, Ying Luo, and Sen-ching S Cheung. Privacy protected image denoising with secret shares. In *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pages 253–256. IEEE, 2012.

[30] S. Ye, Y. Luo, J. Zhao, and S.S. Cheung. Anonymous Biometric Access Control. *EURASIP Journal on Information Security*, 2009, Article ID 865259, 17 pages, 2009.

[31] M. Blanton and P. Gasti. Secure and efficient protocols for iris and fingerprint identification. Technical report, Cryptology ePrint Archive, Report 2010/627, 2010. http://eprint. iacr. org, 2010.

[32] A.R. Sadeghi, T. Schneider, and I. Wehrenberg. Efficient privacy-preserving face recognition. *Information, Security and Cryptology–ICISC 2009*, pages 229–244, 2010.

[33] U. Uludag and A. Jain. Securing fingerprint template: Fuzzy vault with helper data. *Proceedings of CVPR Workshop on Privacy Research In Vision*, 2006.

[34] L. Sweeney. k-anonymity: a model for protecting privacy. In *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, pages 557–570, 2002.

[35] Ying Luo and S Cheung Sen-ching. Privacy information management for video surveillance. In *SPIE Defense, Security, and Sensing*, pages 871207–871207. International Society for Optics and Photonics, 2013.

[36] J. Schiff, M. Meingast, D. Mulligan, S. Sastry, and K. Goldberg. Respectful cameras: Detecting visual markers in real-time to address privacy concerns. In *International Conference on Intelligent Robots and Systems (IROS)*, 2007.

[37] D. Chen, Y. Chang, R. Yan, and J. Yang. Tools for protecting the privacy of specific individuals in video. *EURASIP Journal on Advances in Signal Processing*, 2007:Article ID 75427, 9 pages, 2007. doi:10.1155/2007/75427.

[38] H. Wactlar, S. Stevens, and T. Ng. *Enabling Personal Privacy Protection Preferences in Collaborative Video Observation.* NSF Award Abstract 0534625, http://www.nsf.gov/awardsearch/showAward.do?award Number=0534625.

[39] Y. Deswarte and M. Roy. Privacy-enhancing access control enforcement. In *W3C Workshop on Languages for Privacy Policy Negotiation and Semantics-Driven Enforcement*, 2006.

[40] J. Hotelling. Analysis of a complex of statistical variables into principal components. *J. of Educational Psychology*, 24:417–441, 1933.

[41] Trevor F. Cox and Michael A.A. Cox. *Multidimensional scaling*. Boca Raton : Chapman & Hall, second edition, 2001.

[42] C. Faloutsos and King-Ip Lin. Fastmap: a fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets. In *Proceedings of ACM-SIGMOD*, pages 163–174, May 1995.

[43] J. Bourgain. On lipschitz embedding of finite metric spaces in hilbert space. *Israel Journal of Mathematics*, 52:46–52, 1985.

[44] A. Gionis, P. Indyk, and R. Motwani. Similarity search in high dimneions via hashing. In *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB)*, 1999.

[45] O. Goldreich. *Foundations of Cryptography: Volume 1, Basic Tools*. Cambridge University Press, 2007.

[46] P. Mohanty, S. Sarkar, and R. Kasturi. Privacy and security issues related to match scores. In *Proceedings of Computer Vision and Pattern Recognition Workshop*, pages 162–165, June 2006.

[47] C. Fontaine and F. Galand. A survey of homomorphic encryption for nonspecialists. *EURASIP Journal on Information Security*, 2007, 2007.

[48] A. C. Yao. Protocols for secure computations. In *Proceedings of the 23rd Annual IEEE Symposium on Foundations of computer science*, 1982.

[49] J. Katz and Y. Lindell. *Introduction To Modern Cryptography*. Chapman and Hall, 2008.

[50] W. Aiello, Y. Ishai, and O. Reingold. Priced oblivious transfer: How to sell digital goods. In *Advances in Cryptology – EUROCRYPT*, volume 2045, pages 119–135, 2001.

[51] H. Lipmaa. Verifiable homomorphic oblivious transfer and private equality test. In *Advances in Cryptology – ASIACRYPT*, volume 2894, 2003.

[52] Y. Ishai, J. Kilian, K. Nissim, and E. Petrank. Extending oblivious transfers efficiently. *Advances in Cryptology-CRYPTO 2003*, pages 145–161, 2003.

[53] D. Beaver. Precomputing oblivious transfer. *Advances in CryptologyCRYPT095*, pages 97–109, 1995.

[54] A. C. Yao. How to generate and exchange secrets. In *Proceedings of the 27th Annual IEEE Symposium on Foundations of Computer Science*, pages 162–167, 1986.

[55] Y. Lindell and B. Pinkas. A proof of Yao's protocol for secure two-party computation. In *Electronic Colloquium on Computational Complexity*, volume 11, page 063. Citeseer, 2004.

[56] D. Malkhi, N. Nisan, B. Pinkas, and Y. Sella. Fairplay — a secure two-party computation system. In *USENIX*, 2004. `http://www.cs.huji.ac.il/project/Fairplay`.

[57] V. Kolesnikov and T. Schneider. Improved garbled circuit: Free xor gates and applications. *Automata, Languages and Programming*, pages 486–498, 2008.

[58] M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness thorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the 20th ACM Symposium on the Theory of Computing*, pages 1–10, 1988.

[59] Adi Shamir. How to share a secret. *Communications of the ACM*, 22(1):612–613, 1979.

[60] Josh Benaloh. Secret sharing homomorphisms: keeping shares of a secret secret. In *Proceedings on Advances in cryptology—CRYPTO '86*, pages 251–260, London, UK, UK, 1987. Springer-Verlag.

[61] K. H. Rosen. *Elementary Number Theory.* Addison Wesley, 1988.

[62] N.K. Ratha, J.H. Connell, and R.M. Bolle. Enhancing security and privacy in biometrics-based authentication systems. *IBM Systems Journal*, 40(3):614–634, 2001.

[63] S. Hoque, M. Fairhurst, G. Howells, and F. Deravi. Feasibility of generating biometric encryption keys. *Electronics Letters*, 41(6):309–311, 2005.

[64] E. N. Newton, Latanya Sweeney, and B. Main. Preserving privacy by de-identifying face images. *IEEE transactions on Knowledge and Data Engineering*, 17(2):232–243, February 2005.

[65] V. Ciriani, S. De Capitani di Vimercati, S. Foresti, and P. Samarati. k-anonymity. *Secure Data Management in Decentralized Systems*, 2007.

[66] O. Goldreich. *Foundations of Cryptography: Volume II Basic Applications.* Cambridge, 2004.

[67] B. Goethals, S. Laur, H. Lipmaa, and T. Mielikainen. On private scalar product computation for privacy-preserving data mining. In *The 7th Annual International Conference in Information Security and Cryptology (ICISC2004)*, volume 3506, pages 104–120, 2004.

[68] M. Naor and B. Pinkas. Oblivious polynomial evaluation. *SIAM J. Comput*, 35(5):1254–1281, 2006.

[69] Y.-C. Chang and C.-J. Lu. Oblivious polynomial evaluation and oblivious neural learning. *Theoretical Computer Science*, 341:39–54, 2005.

[70] I. Damgard, M. Geisler, and M. Kroigard. Homomorphic encryption and secure comparison. *International Journal of Applied Cryptography*, 1(1):22–31, 2008.

[71] Marc Fischlin. A cost-effective pay-per-multiplication comparison method for millionaires. *Lecture Notes in Computer Science*, 2020:457–472, 2001.

[72] E. Kiltz, P. Mohassel, E. Weinreb, and M. Franklin. Secure linear algebra using linearly recurrent sequences. *LECTURE NOTES IN COMPUTER SCIENCE*, 4392:291–305, 2007.

[73] R. Cramer and I. Damgaard. Secure distributed linear algebra in constant number of rounds. In *Proceedings 21st Annual IACR CRYPTO'01*, volume 2139 of *LNCS*, pages 119–136. Springer-Verlag, 2001.

[74] B. Schoenmakers and P. Tuyls. Efficient binary conversion for paillier encrypted values. *EUROCRYPT*, 4004:522–537, 2006.

[75] G. Jagannathan, K. Pillaipakkamnatt, and R. N. Wright. A new privacy-preserving distributed k-clustering algorithm. *Proceedings of the Sixth SIAM International Conference on Data Mining*, 2006.

[76] M. C. Doganay, T. B. Pedersen, Y. Saygin, E. Savaş, and A. Levi. Distributed privacy preserving k-means clustering with additive secret sharing. *Proceedings of the 2008 international workshop on Privacy and anonymity in information society*, pages 3–11, 2008.

[77] S. Samet and A. Miri. Privacy preserving id3 using gini index over horizontally partitioned data. *Computer Systems and Applications, 2008. AICCSA 2008. IEEE/ACS International Conference on*, pages 645–651, 2008.

[78] J. Zhan. Privacy-preserving decision tree classification in horizontal collaboration. *Security of Information and Networks: Proceedings of the First International Conference on Security of Information and Networks (Sin 2007)*, 2008.

[79] Yehuda Lindell and Benny Pinkas. Privacy preserving data mining. *Journal of Cryptology*, 15(3):177–206, 2002.

[80] J. Vaidya, H. Yu, and X. Jiang. Privacy-preserving svm classification. *Knowledge and Information Systems*, 14(2):161–178, 2008.

[81] C. Orlandi, A. Piva, and M. Barni. Oblivious neural network computing via homomorphic encryption. *EURASIP Journal on Information Security*, Volume 2007, 2007.

[82] R. Wright and Z. Yang. Privacy-preserving bayesian network structure computation on distributed heterogeneous data. *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 713–718, 2004.

[83] S.-C. Cheung and T. Nguyen. Secure signal processing between distrusted network terminals. *EURASIP Journal on Information Security*, 2007. http://www.hindawi.com/GetArticle.aspx?doi=10.1155/2007/51368.

[84] C. Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the 41st annual ACM symposium on Theory of Computing*, pages 169–179, 2009.

[85] M. Cooney. Ibm touts encryption innovation. *Computer World*, June 25 2009.

[86] B. Schneier. Homomoprhic encryption breakthrough. In *Schneier on Security*. http://www.schneier.com/blog/archives/2009/07/homomorphic_enc.html, 2009.

[87] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. *Proceedings of International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT ?9)*, vol. 1592:223–238, May 1999.

[88] Z. Erkin, A. Piva, S. Katzenbeisser, R.I.L. Lagendijk, J. Shokrollahi, G. Neven, and M. Barni. Protection and retrieval of encrypted multimedia content: When cryptography meets signal processing. *EURASIP Journal on Information Security*, 2007, 2007.

[89] T. Bianchi, A. Piva, and M. Barni. Discrete cosine transform of encrypted images. In *Proceedings of IEEE International Conference on Image Processing*, 2008.

[90] B. Pinkas, T. Schneider, N. Smart, and S. Williams. Secure two-party computation is practical. *Advances in Cryptology–ASIACRYPT 2009*, pages 250–267, 2009.

[91] E. Barker, W. Burr, A. Jones, T. Polk, S. Rose, M. Smid, and Q. Dang. Recommendation for key management. *NIST special publication*, 2009.

[92] G. V. Lioudakis et al. A middleware architecture for privacy protection. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 51(16):4679–4696, November 2007.

[93] D.-A. Fidaleo, H.-A. Nguyen, and M. Trivedi. The networked sensor tapestry (nest): a privacy enhanced software architecture for interactive analysis of data in video-sensor networks. In *VSSN '04: Proceedings of the ACM 2nd international workshop on Video surveillance & sensor networks*, pages 46–53, New York, NY, USA, 2004. ACM Press.

[94] S.-C. Cheung, M. V. Venkatesh, J. Paruchuri, J. Zhao, and T. Nguyen. Protecting and managing privacy information in video surveillance systems. In A. Senior, editor, *Protecting Privacy in Video Surveillance*. Springer, 2009.

[95] U. Uludag, S. Pankanti, S. Prabhakar, and A.K. Jain. Biometric cryptosystems: issues and challenges. *Proceedings of the IEEE*, 92(6):948–960, 2004.

[96] F. Hao, R. Anderson, and J. Daugman. Combining cryptography with biometrics effectively. *IEEE Transactions on Computers*, 55(9):1081–1088, 2006.

[97] A. Cavoukian and A. Stoianov. *Biometric encryption: a positive-sum technology that achieves strong authentication, security and privacy*. Information and Privacy Commissioner, Ontario, 2007.

[98] J.P. Linnartz and P. Tuyls. New shielding functions to enhance privacy and prevent misuse of biometric templates. In *Audio-and Video-Based Biometric Person Authentication*, pages 1059–1059. Springer, 2003.

[99] L. Masek and P. Kovesi. Matlab source code for a biometric identification system based on iris patterns. Technical report, The School of Computer Science and Software Engineering, The University of Western Australia, 2003.

[100] I. Damgard, M. Geisler, and M. Kroigard. Homomorphic encryption and secure comparison. *International Journal of Applied Cryptography*, 1(1):22–31, 2008.

[101] M. Barni, J. Guajardo, and R. Lazzeretti. Privacy preserving evaluation of signal quality with application to ecg analysis. In *Information Forensics and Security (WIFS), 2010 IEEE International Workshop on*, pages 1–6. IEEE, 2010.

[102] V. Kolesnikov, A.R. Sadeghi, and T. Schneider. Improved garbled circuit building blocks and applications to auctions and computing minima. *Cryptology and Network Security*, pages 1–20, 2009.

[103] R. Lazzeretti and M. Barni. Division between encrypted integers by means of garbled circuits. *The 2011 IEEE Intl. Workshop on Information Forensics and Security (WIFS'11)*, 2011.

[104] V. Kolesnikov, A.R. Sadeghi, and T. Schneider. How to Combine Homomorphic Encryption and Garbled Circuits. *Signal Processing in the Encrypted Domain*, pages 100–121, 2009.

[105] Y. Li, M. Savvides, and T. Chen. Investigating useful and distinguishing features around the eyelash region. In *2008 37th IEEE Applied Imagery Pattern Recognition Workshop*. IEEE, 2008.

[106] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, et al. A view of cloud computing. *Communications of the ACM*, 53(4):50–58, 2010.

[107] O. Catrina and S. De Hoogh. Improved primitives for secure multiparty integer computation. *Security and Cryptography for Networks*, pages 182–199, 2010.

[108] T. Tan and Z. Sun. Casia-irisv3. Technical report, Chinese Academy of Sciences Institute of Automation, http://www.cbsr.ia.ac.cn/IrisDatabase.htm, 2005.

[109] J. Bethencourt. *Paillier Library*. UC Berkeley, http://acsc.csl.sri.com/libpaillier/.

[110] J.L. Devore. Probability and Statistics for Engineering and the Science, Brooks/Cole Pub. *Co., Monterey, California*, 704, 1991.

[111] J.B. Wilmer et al. Human face recognition ability is specific and highly heritable. *Proceedings of the National Academy of Sciences*, 2010.

[112] A.K. Jain, S. Prabhakar, and S. Pankanti. On the similarity of identical twin fingerprints. *Pattern Recognition*, 35(11):2653–2663, 2002.

[113] A.W.K. Kong, D. Zhang, and G. Lu. A study of identical twins' palmprints for personal verification. *Pattern Recognition*, 39(11):2149–2156, 2006.

[114] D. Filho and P. Barreto. Demonstrating data possession and uncheatable data transfer. In *Cryptology ePrint Archive*. Report 2206/150, 2006.

[115] M Krohn, M. Freedman, and D. Mazieres. On-the-fly verification of rateless erasure codes for efficient content distribution. In *Proc. IEEE Symposium on Security and Privacy*, pages 226–240, 2004.

[116] X. Yu and N. Babaguchi. Privacy preserving: Hiding a face in a face. In *ACCV*, pages 651–661, 2007.

[117] M. Vijay Venkatesh, S.-C. Cheung, and J. Zhao. Efficient object-based video inpainting. *Pattern Recognition Letters : Special issue on Video-based Object and Event Analysis*, 2008.

[118] Frdric Dufaux and Touradj Ebrahimi. Scrambling for video surveillance with privacy. *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*, page 160, 2006.

[119] K. Martin and K. N. Plataniotis. Privacy protected surveillance using secure visual object coding. *To appear IEEE Transactions on Circuits and Systems for Video Technology*, 2008.

[120] ITU-T Recommendation H.263 Version 2. *Video Coding for Low Bitrate Communication Version 2*, 1998.

[121] J. Zhao, S.-C. Cheung, and T. Nguyen. Optimal camera network configurations for visual tagging. *IEEE Journal on Selected Topics in Signal Processing*, 2(4):464–479, Aug. 2008.

[122] W. Gasarch. A survey on private information retrieval. *The Bulletin of the EATCS*, 82:72–107, 2004.

[123] A. Uhl and A. Pommer. *Image and Video Encryption: From Digital Rights Management to Secured Personal Communication*, volume 15. Springer, 2004.

Vita

Ying Luo

EDUCATION

- **Ph.D.** **Department of Electrical and Computer Engineering**
  University of Kentucky (GPA:3.9/4) **08/2014**
  Dissertation Title: *Efficient Anonymous Biometric Matching in Privacy-aware Environments*
  Research Advisor: Dr. Sen-ching "Samson" Cheung
  Committee: Sen-ching Cheung, Laurence G. Hassebrook, Kevin Donohue, Jun Zhang

- **M.S.** **College of Computer Science**
  Sichuan University, Chengdu, Sichuan, China **07/2006**
  Thesis Title: *The Research and Application of Fuzzy Information Systems Based on Rough Set and Inclusion Degree*
  Advisor: Dr. Hongwei Zhang

- **B.S.** **College of Computer Science**
  Sichuan University, Chengdu, Sichuan, China **07/2003**
  Thesis Title: *The Analysis and Design of Warehouse Management Information System*

PROFESSIONAL EXPERIENCE

09/2010 – present **Graduate Research Assistant.**
Center for Visualization and Virtual Environments, Univ. Of Kentucky

08/2011 – 11/2011 **Visiting Student**
Department of Information Engineering, Univ. of Siena, Siena, Italy
Supervisor: Dr. Mauro Barni

08/2007 – 05/2010 **Teaching Assistant**.
Department of Electrical and Computer Engineering, Univ. Of Kentucky
Courses assisted: Capstone Design Course for senior students,
Electrical Circuits and Electronics

05/2008 – 08/2008 **Graduate Research Assistant.**
Center for Visualization and Virtual Environments, Univ. Of Kentucky

07/2006 – 07/2007 **Instructor**.
Department of Information Engineering
Sichuan Technology and Business College, Chengdu, China
Courses taught: Data Structure, Windows Forms development with C# language, Computer English, and Computer Application Foundation

10/2003 – 07/2005 **Software Engineer & Customer Service Representative**
Jin Cai Science & Technology Company, Chengdu, Sichuan, China

SCHOLARSHIPS AND AWARDS

- Travel grant for GREPSEC workshop for women and underrepresented groups interested in computer security research (May 2013)
- IEEE Signal Processing Society Travel Grant for ICIP2012 (September, 2012)
- Student Travel Grant Award for ICME2010 (July 2010)
- Sichuan University Second-class Scholarship (2004-2005)

- Sichuan University Second-class Scholarship (2001-2002)
- Sichuan University First-class Scholarship and Award for Excellent student (2000-2001)
- Sichuan University Third-class Scholarship and Award for Excellent student (1999-2000)

PROJECT EXPERIENCE

- **Video Interface for Behavioral Evaluation (VIBE)** 01/2014-present
  (Sponsored by NSF Award #1444022 since 07/2014)
  - ✓Assisted in writing grant proposal.
  - ✓Take the role of the entrepreneurial lead to transit the VIBE technology into market.
  - ✓Design an audiovisual recording and coding system for clinical use, specially used to capture human behaviors in naturalistic environments such as home and school.
  - ✓Combine state-of-the-art multi-modal surveillance, privacy protection, and event coding technologies.
  - ✓Investigate the possibility of commercializing VIBE system.

- **Privacy Protection of Multimedia Processing** 05/2007-present
  (Sponsored by NSF Award #1018241 since 09/2010)
  - ✓Assisted in writing grant proposal.
  - ✓Investigated the combination of signal processing techniques and secure cryptographical primitives in breaking the "efficiency barrier" of classical Secure Multiparty Computation (SMC) schemes.
  - ✓Proposed the Anonymous Biometric Access Control (ABAC) system to protect user anonymity.
  - ✓Proposed the k-Anonymous Quantization (kAQ) framework that provides an effective and secure tradeoff of privacy and complexity.
  - ✓Applied ABAC into privacy-aware video surveillance.
  - ✓Compared the computational and communication complexity of different primitives, such as homomorphic encryption, garbled circuits, and secret sharing.

- **Visualization for Training and Simulation in Night Environments** 05/2013-09/2013
  (Sponsored by Lockheed Martin)
  - ✓Integrated 3D point-cloud data from multiple Kinects and displayed them in a single 3D viewer in real-time.
  - ✓Tracked the skeleton of the detected individual from Kinects.
  - ✓Synchronized multiple Kinects with a local set-up NTP time server.

- **Customer Relationship Management & Distribution Resource Planning System** 09/2003-08/2005
  - ✓Processed the assessment ranking in the fuzzy information system.
  - ✓Designed an optimal ranking algorithm by introducing Genetic Algorithm to solve the problem caused by mass data.

PUBLICATIONS

Conference

[1] Ju Shen, Wanxin Xu, Ying Luo, Po-Chang Su, and Sen-ching S. Cheung. "Extrinsic Calibration for Wide-baseline RGB-D Camera Network", submitted to **IEEE International Conference on Image Processing (ICIP 2014)**

[2] Zhaohong Wang, Ying Luo, and Sen-ching S. Cheung. "Efficient Multi-party Computation with Collusion-deterred Secret Sharing", **In IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2014)**, Florence, Italy, May 4-9.

[3] Ying Luo, and Sen-ching S. Cheung. "Privacy information management for video surveillance", **In Proc. SPIE 8712, Biometric and Surveillance Technology for Human and Activity Identification X**, 871207 (May 31, 2013), doi:10.1117/12.2015999, http://dx.doi.org/10.1117/12.2015999, 7 pages.

[4] Ying Luo, Sen-ching S. Cheung, Tommaso Pignata, Riccardo Lazzeretti, Mauro Barni, "An Efficient Protocol for Private Iris-code Matching by Means of Garbled Circuits", **Invited paper in IEEE International Conference on Image Processing (ICIP 2012),** Orlando, Florida, USA, Sept. 30-Oct. 3, 2012, pp. 2653-2656

[5] S. M. SaghaianNejadEsfahani, Ying Luo, and Sen-ching S. Cheung, "Privacy Protected Image Denoising with Secret Shares", **In IEEE International Conference on Image Processing (ICIP 2012),** Orlando, Florida, USA,

Sept. 30-Oct. 3, 2012, pp. 253-256

[6] Ying Luo, Shuiming Ye, and Sen-ching S. Cheung, "Anonymous Subject Identification in Privacy-aware Video Surveillance", **Invited paper in IEEE International Conference on Multimedia & Expo (ICME2010)**, Singapore, July 2010, pp. 83-88

[7] Ying Luo, Sen-ching S. Cheung, and Shuiming Ye, "Anonymous Biometric Access Control based on Homomorphic Encryption", In **IEEE International Conference on Multimedia & Expo (ICME2009)**, Cancun, Mexico, June 2009, pp. 1046-1049

Journal

Published

[1] Shuiming Ye, Ying Luo, Jian Zhao and Sen-ching S. Cheung. "Anonymous Biometric Access Control", In **EURASIP Journal on Information Security: special issue on privacy protection in multimedia systems**, vol. 2009, Article ID 865259, 17 pages, 2009.

[2] Ying Luo, Hongwei Zhang, Dan Li, and Xiang Zhong, "Knowledge Discovery of Fuzzy Rules and its Application in CRM", In **Computer Engineering & Applications**, Beijing, China,2005, Vol. 41, No. 35, pp. 229-232;

[3] Dan Li, Hongwei Zhang, Ying Luo, "MultiObjective Many-Person Decision with Consistent Measurement and its Application in Credit-Evaluating", In **Computer Engineering & Applications**, Beijing, China, 2006, Vol. 42, No. 15, pp. 193-197;

[4] Xiang Zhong, Hongwei Zhang, Huairong Weng and Ying Luo, "Discovery of Sets Information System and Grey Theory and theirs Application in DRP", In **Computer Applications**, Chengdu, China, 2005, Vol. 25, No. 10, pp. 2447-2449.

In preparation

[1] Ying Luo, and Sen-Ching Samson Cheung "Auditable Video Surveillance", to be submitted to **IEEE Transactions on Information Forensics and Security**.

[2] Ying Luo, and Sen-Ching Samson Cheung, "Privacy-preserving image processing with secure multiparty computation"

Book chapter

[1] Ying Luo, Sen-ching S. Cheung, and Shuiming Ye, 2012. "Anonymity in Video Surveillance System". To appear in **Intelligent Multimedia Surveillance: Current Trends and Research**, edited by P. Atrey, M. Kankanhalli, and Andrea Cavallaro, Springer.

[2] Jithendra Parachuri, Ying Luo and Sen-ching S. Cheung, 2012. "Managing Privacy Information In Video Surveillance Systems" To appear in **Effective Surveillance for Homeland Security: Balancing Technology and Social Issues**, edited by CRC Press

PROFESSIONAL ACTIVITIES

- Member of IEEE and SPIE
- Reviewer for
  - IEEE Transactions on Information Forensics & Security
  - IEEE Signal Processing Letters
  - IEEE Transactions on Multimedia
  - IEEE Transactions on Industrial Informatics
  - Signal Processing: Image Communication
  - EURASIP Journal on Information Security
  - IET Information Security
- Invited speaker of the Keeping Current Seminar in department of Computer Science at University of Kentucky: *Privacy-aware Biometric Matching based on Secure Multiparty Computation*. Apr 24, 2013