Semba: Secure multi-biometric authentication

SEMBA: SEcure Multi-Biometric Authentication

Giulia Droandi, Tommaso Pignata
and Mauro Barni
Department of Information Engineering and Mathematics
University of Siena, Siena, Italy
Email: {giulia.droandi,pignata.tommaso}@gmail.com
barni@diism.unisi.it

Riccardo Lazzeretti
Department of Computer, Control, and
Management Engineering ”Antonio Ruberti”
Sapienza University of Rome, Rome, Italy.
Email: lazzeretti@diag.uniroma1.it

***Abstract*—Biometrics security is a dynamic research area spurred by the need to protect personal traits from threats like theft, non-authorised distribution, reuse and so on. A widely investigated solution to such threats consists in processing the biometric signals under encryption, to avoid any leakage of information towards non-authorised parties. In this paper, we propose to leverage on the superior performance of multimodal biometric recognition to improve the efficiency of a biometric-based authentication protocol operating on encrypted data under the malicious security model. In the proposed protocol, authentication relies on both facial and iris biometrics, whose representation accuracy is specifically tailored to trade-off between recognition accuracy and efficiency. From a cryptographic point of view, the protocol relies on SPDZ [1], [2]. Experimental results show that the multimodal protocol is faster than corresponding unimodal protocols achieving the same accuracy.**

I. Introduction

In the digital and increasingly interconnected world we live in, establishing individuals' identity is a pressing need. For this reason, in the last decades, we have seen an increasing interest in biometric-based recognition systems. Biometric recognition can be split into two main categories: authentication and identification. In the first scenario, also referred to as *verification*, the user is interested in demonstrating that he/she is who he/she claims to be, while in the second one, the goal is to determine the identity of the user submitting the biometric template among those *known by the system*. Usually, in both verification and identification protocols, a single biometric trait is used to extract a feature vector. The feature vector then is matched with one or more templates stored in the system database. In this work we focus on an authentication protocol.

More recently, the security of biometric systems has become a very active research area, due to the necessity of impeding newly emerging cybercrimes like identity theft, privacy violation, unauthorized access to sensitive information and so on [3]. Protocols allowing to process encrypted biometric signals without decrypting them are among the most widely studied solutions to enhance the security of biometric systems [4], [5]. According to such an approach, verification or identification is carried out by the system by relying only on encrypted biometric templates, thus avoiding the risk that sensitive information is leaked during the protocol.

The possibility of processing a comparing encrypted biometric templates relies on a number of cryptographic tools [6], [7], [8], [9], broadly referred to as Multi-Party Computation (MPC) [10]. Generally speaking, MPC protocols can be classified according to the adopted security model. The most common distinction considers protocols which are secure only against *semi-honest* adversaries, and those which can be proven to be secure also against *malicious adversaries*. To be specific, according to the *semi-honest* model, all the parties execute the protocol without deviating from it, but meanwhile they try to obtain as much information as possible about the other parties' data. Protocols developed in the semi-honest model are very efficient and, for this reason, are adopted in the majority of the works proposed so far [4], [5]. On the contrary, in the *malicious* model, the parties can arbitrarily deviate from the protocol in their attempt to get access to sensitive information. While security against malicious parties is desirable, in many real world applications, the resulting protocols have a very high complexity and their use in real systems is often impractical. The great majority of the attempts made so far to devise efficient biometric recognition protocols in the malicious setting, focused on the development and use of innovative and efficient MPC and cryptographic primitives. A much less investigated approach consists in the adoption of biometric recognition protocols which are better suited to be implemented in a MPC framework. Yet, as highlighted in [5], working on the signal processing side of the problem may help to reduce significantly the complexity of the resulting MPC protocol, e.g. by efficiently trading off between accuracy and complexity.

A. Contribution

In this work, we follow the above strategy and present SEMBA: a SEcure Multi-Biometric Authentication protocol which achieves a better trade-off between efficiency and accuracy with respect to the single modality subsystems composing it. This represents a major departure from most works on multimodal biometric systems, in which the availability of multiple biometric modalities is exploited to decrease interclass variability and improve intra-class similarity in the presence of acquisition noise and any other kind of distortion [11]. In this framework, the main contributions of the paper are the following: *i)* we design a multimodal biometric system that combines face and iris templates and that can be easily implemented by relying on Secure Multiparty Computation protocols; *ii)* we propose a privacy preserving multi-biometric authentication protocol secure against a malicious party. SEMBA is based on the SPDZ tool [1], [2] and discloses only the final binary decision; *iii)* we compare our multi-biometric protocol with single biometric protocols, showing that by using a properly simplified representation of the two biometric traits, backed by a rigorous signal processing analysis, the multimodal protocol can reach the same accuracy of the corresponding single-modality systems based on more accurate - and more complicated - representations of iris and face templates, but with significantly lower computational complexity. In particular, SEMBA obtains the same accuracy of the stand alone iris authentication protocol described in [12]. Of course, system designers could also decide to exploit the superior performance allowed by multimodal authentication to improve authentication accuracy with the same complexity of the single modality protocols.

II. Prior works

In the last years, many cryptographic tools, including oblivious transfer [6], homomorphic encryption [8], [13], and garbled circuits [7], have been used for privacy protection of biometrics templates. In most works, such tools are used in such a way to achieve security in the semi-honest model. Many privacy preserving authentication protocols have been proposed in the literature making use of a wide variety of biometric traits. Since, in this work, we present a privacy preserving multibiometric authentication protocol based on face and iris, we focus on the state of art relative to those biometries, then we discuss the few works achieving privacy protection in the malicious model and finally we discuss the main characteristics of multimodal (or *fusion*) biometric systems.

A. Biometric recognition in the semi-honest model

As pointed in [4], [5], many prior works on biometric recognition are designed to be secure against semi-honest adversaries. Some examples are [14], [15] for face recognition and [16], [17], [18] for iris recognition. In order to guarantee security, the above papers are based on many cryptographic techniques such as Pailler additive homomorphic cryptosystem (HE), Oblivious Transfer (OT) or Garbled Circuit (GC). For efficiency reasons, such protocol are mainly based on the eigenface [19] and iriscode [20] representation of iris and face respectively. More accurate protocols exist, but their privacy preserving implementation has such a high complexity to make them impractical.

B. Biometric recognition in malicious setting

There are few works on privacy preserving biometric authentication secure under a malicious model. Kantarcioglu and Kardes [21] present a way to implement some primitives, specifically the dot product and equality check, in the malicious model, by also analyzing the corresponding computational cost. Even if this work is not directly related to biometrics, the proposed solutions can be adapted to them. In [22], Abidin presents a general framework for biometric authentication that uses a homomorphic encryption scheme to evaluate the distance between two encrypted biometric templates. In his work, Abidin proves security against malicious attacks, but does not provide results about the practical implementation of the protocol. In [23], Pathak and Raj present two speech-based authentication protocols. The first one is an interactive protocol based on Pailler cryptosystem which is secure against a semi-honest adversary, the second one is a non-interactive protocol based on BNG [24] cryptosystem, which allows to perform an arbitrary number of additions and one multiplication between ciphertexts, and is secure against malicious attacks. In both cases the output is a probability value and the client checks if such a value is equal to zero or not in the plain domain. From the tests and the analysis reported in [23] it is clear that the interactive protocol is more efficient than the malicious one.

A large number of approaches [25], [26], [27], [28], [29] have been proposed to make Yao's garbled circuit techniques secure in the malicious model through Zero Knowledge proof, cut and choose, or other techniques. Such approaches can also be used for biometric authentication protocols, but their complexity is so high to make them impractical. Gasti et al. [30] propose a lightweight biometric authentication protocol based on simple garbled circuits and secure against malicious adversaries by relying on an untrusted third party (the cloud). The goal of the protocol is to minimize the amount of computation performed by the biometric owner's device (a smartphone), while also reducing the protocol execution time and without the necessity to rely on cut-and-choose techniques. In the protocol, the biometric owner acts as circuit constructor, the cloud as circuit evaluator, while the server verifies the correctness of the circuit. The approach is secure against colluding biometric owner and cloud, but not against colluding server and cloud.

C. Multimodal biometric recognition

Given the recent technological advances, novel devices are often equipped with numerous sensors, opening the way to multi-biometric authentication. In [31] Ross, Nandakumar, and Jain present an overview of the possible fusion scenarios and their applications in real life. For our protocol we choose a *multimodal* system, that combines information from face and iris.

Biometric signals are usually processed in four stages. First a sensor captures the traits of an individual as a raw biometric data. Second, raw data are processed and a compact representation of the physical traits, called *features*, is extracted. Then the feature template is matched with the templates stored in a database. Finally the matching score is used to determine an identity or to validate a claimed identity. Information can be merged at any time during a multi-biometric recognition protocol. For further information about this topic readers may refer to [31]. The choice of fusion depends on the intended application, its specific characteristics and the multiparty computation tool chosen to guarantee privacy. Fusing the biometric signals at an early stage results in a higher accuracy of the protocol at the expenses of a higher complexity. For this reason, the most used approach, and the one we use in this paper, is score level fusion, whereby the match scores from each biometric trait involved in the process are combined to obtain the final result. Score level fusion combines good accuracy and relatively easy implementation.

To the best of our knowledge, Gomez-Barrero et al. [32] have proposed the only previous work on multibiometric privacy protection operating in the encrypted domain. In their work, the authors present a general framework for multibiometric template protection based on Pailler cryptosystem, in which only encrypted data is handled. The authors examine the outcome of the fusion of on-line signature and fingerprints, at three different levels of fusion: feature, score and decision levels. The system presented by Gomez et al. has a low computational cost (only one decryption on the server side and no encryptions at verification time), moreover they obtained a good accuracy (EER = 0.12%), with a required time for a single comparison of about $5 \cdot 10^{-4}$s. A drawback with the system described in [32], is that comparison is carried out on plain data by the server, thus introducing a breach into the security of the system. On the contrary, in our work we implement also the final comparison step within the SPDZ framework, to prevent any security loss, even if this choice has a non-negligible cost in terms of complexity (see Section VI). As a further difference, in [32] an Euclidean distance computation (in case of two-modal system) requires $M \cdot F + 2$ exponentiations, where M is the number of enrolled samples and F the feature's total number considering all the modalities. In our work, instead, thanks to the SPDZ system and to the possibility of using integer numbers, we need only k (the length of the feature vector) squares, one of our most expensive operations.

III. Tools

In this section, we present the cryptographic and biometrical tools used in our protocol.

A. *Cryptographic tools: SPDZ system*

Damgård et al. [1], [2] proposed the MPC framework named SPDZ, a two - or multi-party - computation protocol secure against an active adversary corrupting up to $n-1$ of the n players. This method uses multiplicative triples generated offline by using Somewhat Homomorphic Encryption (SHE) to efficiently perform online secret sharing operations.

We assume the computation is performed over a fixed finite field $\mathbb{F}_p$ of characteristic p; where p is a prime number. Each player P_i has an uniform share $\alpha_i \in \mathbb{F}_p$ of a secret key α such that $\alpha = \sum_{i=1}^{n} \alpha_i \mod p$ (in the following we omit the indication of the modulus operation for simplicity). In this paper we focus on secure two-party computation protocols, then $n = 2$ and $\alpha = \alpha_1 + \alpha_2$. An item $a \in \mathbb{F}_p$ is $\langle\cdot\rangle$-shared if the player P_i holds a tuple $\langle a_i, \gamma(a)_i\rangle$ such that $a = a_1 + a_2$ and $\gamma(a) = \gamma(a)_1 + \gamma(a)_2$. In other words, a_i and $\gamma(a)_i$ are additive secret shares of a and $\gamma(a)$. The value $\gamma(a)$ represents the Message Authentication Code (MAC) of a. Any operation involving some variables is also performed on their MAC, so that, at the end of the protocol, the MAC is checked before revealing the outcome. If one of the parties has a different MAC from the others, the procedure aborts. During the description of the protocol, we say that a $\langle\cdot\rangle$ - shared value is *partially opened* if each party reveals to the other one the value a_i but not the associated $\gamma(a)_i$.

An SPDZ protocol can be divided into two major phases. The preprocessing phase, sometimes referred to as the offline phase, where the system is set up, and the online phase, where the actual computation is performed. In the offline phase, parties generate a public key and a shared secret key for the SHE scheme. Then, relying on the homomorphic properties of the SHE, the pre-processing protocol generates α and α's shares, input shares, shares of tuples for multiplications and squares, and the random share values necessary to evaluate the comparison [2]. Finally each party decrypts his set of preprocessed data by using his secret key share. In this paper, we assume that the generation of tuples and inputs has been already made in the encrypted domain before the protocol starts and we focus our efforts on the analysis of the online part of the system.

TABLE I. LINEAR OPERATION IN SPDZ. WITH $\langle a\rangle$ WE INDICATE THE PAIR $\langle a, \gamma(a)\rangle$, $\langle a\rangle_i$ INDICATES THE PAIR $\langle a_i, \gamma(a)_i\rangle$.

operation	party 1	party 2
$\langle a\rangle + \langle b\rangle$	$\langle a\rangle_1 + \langle b\rangle_1$	$\langle a\rangle_2 + \langle b\rangle_2$
$\langle a\rangle - \langle b\rangle$	$\langle a\rangle_1 - \langle b\rangle_1$	$\langle a\rangle_2 - \langle b\rangle_2$
$\alpha \cdot \langle a\rangle$	$\alpha \cdot \langle a\rangle_1$	$\alpha \cdot \langle a\rangle_2$
$c + \langle a\rangle$	$c + \langle a\rangle_1$	$\langle a\rangle_2$

By using SPDZ, linear operations, such as additions and scalar multiplications (see Table I), can be performed on the $\langle\cdot\rangle$-shares without interaction; while products between ciphertexts and comparisons need data transmission and proper sub-protocols. Products and square operations are evaluated through interactive protocols that use multiplication triples generated during the preprocessing phase. Due to lack of space we refer to [1], [2] for implementation details. Each multiplication requires two transmissions from each party to the other, while each square operation requires only one transmission.

1) Comparison: Here we show how to compute the outcome of a secure comparison $x < y$, for any two elements $x, y \in \mathbb{F}_p$, according to the protocol proposed in [33], that has the lowest computational complexity among all the secure comparison protocols proposed so far.

The comparison computation is based on the observation that $\langle x < y\rangle$ is determined by the truth values of $\langle x < \frac{p}{2}\rangle$, $\langle y < \frac{p}{2}\rangle$, and $\langle (x-y) \mod p < \frac{p}{2}\rangle$, where $\langle x < y\rangle$ indicates the share values of the outcome of $x < y$. By choosing p so large that both inputs are lower than $\frac{p}{2}$, it is sufficient to evaluate only $\langle (x-y) \mod p < \frac{p}{2}\rangle$.

Given $z = x - y$, then $\langle x < y\rangle$ can be easily computed as $1 - \langle z < \frac{p}{2}\rangle$. We observe that if $z > \frac{p}{2}$ then $2z > p$. Since we work on $\mathbb{F}_p$, we have that $2z \mod p = 2z - p$ and it is odd; else if $z < \frac{p}{2}$ than $2z < p$ and it will be even because we do not need any modular operation. Therefore, to establish if z is larger or smaller than $\frac{p}{2}$ we need to determine only the last significant bit of $2z$. To compute the last significant bit of $2z$, we use a value $\langle r\rangle$ shared by parties both as integer and as a bit array. The value r along with its bit decomposition are pre-computed off line. We indicate as $r_0 r_1 \ldots r_\ell$ the bits of r and with $\langle r_i\rangle$ their shared values.

First of all we compute $\langle s\rangle = \langle 2z + r\rangle$, then s is partially opened. If $s < p$ then the last significant bit of $2z$ is equal to $s_0 \oplus r_0$, otherwise it is equal to $1 - (s_0 \oplus r_0)$.

Since we work in the field $\mathbb{F}_p$, $s < p$ iff $s < r$. By recalling that s is known to both parties, we can easily obtain a $\langle\cdot\rangle$-share of δ, the truth value of $\langle s < r\rangle$ (i.e. $\langle\delta\rangle = \langle s < r\rangle$) working on the bits of s and on the shared bits of r. Then we use the following procedure to calculate $\langle\delta\rangle = \langle s < r\rangle$. nosep,noitemsep

If $s_0 = 0$ then $\langle\delta\rangle = \langle r_0\rangle$ else $\langle\delta\rangle = \langle 0\rangle$.

```
For all $i < \ell - 1$
    if $s_i = 0$ then $\langle\delta\rangle = \langle r_i\rangle + \langle\delta\rangle \cdot \langle 1 - r_i\rangle$
    else $\langle\delta\rangle = \langle r_i\rangle \cdot \langle\delta\rangle$.
```

Now $\langle z < \frac{p}{2}\rangle$ can be easily calculated as

$$\langle\delta \oplus s_0 \oplus r_0\rangle = \langle\delta\rangle - \langle s_0 \oplus r_0\rangle - \langle\delta\rangle \cdot \langle s_0 \oplus r_0\rangle \qquad (1)$$

Since s_0 is known in our implementation,

$$\langle\delta \oplus s_0 \oplus r_0\rangle = \begin{cases} \langle\delta\rangle + \langle r_0\rangle - 2 \cdot \langle\delta\rangle \cdot \langle r_0\rangle & \text{if } s_0 = 0, \\ 1 + 2\langle\delta\rangle \cdot \langle r_0\rangle - \langle r_0\rangle - \langle\delta\rangle & \text{if } s_0 = 1. \end{cases} \qquad (2)$$

Complexity. This protocol requires one multiplication for each iteration plus one for the last step in (1). Therefore the complexity depends on the bit length of r and it is equal to ℓ multiplications, which require 2ℓ transmissions.

B. Biometrics tools

In our work we compare two different authentication systems: an iris authentication protocol and a multimodal system relying on the fusion at the score level of iriscode and eigenfaces. Both protocols are developed in the encrypted domain by relying on the SPDZ framework. Different and more accurate protocols exist, but their privacy preserving implementation have such a high complexity to make them impractical. In the following subsections, we first describe the standalone iris (Section III-B1) and face (Section III-B2) protocols in the plain domain, then the main characteristics of a general multimodal recognition protocol (Section III-B3).

1) Iris recognition: In our implementation, we use the iriscode template proposed for the first time by Daugman in [20] and then modified by Masek in [12]. The description of the entire extraction process is out of the scope of this paper, therefore we limit the presentation to the details which are relevant for the current work.

An iriscode is a bit vector of length N depending on the radial r and angular θ resolutions used during template extraction. The extraction process outputs also a bitwise noise mask. The noise mask represents the regions of the iris altered by noise, e.g. by eyelashes end eyelid. To compare two templates (the query and the probe enrolled in the database) the authentication process relies on a weighted Hamming distance, where the weights depend on the noise mask bits. In this way only significant bits are used to calculate the distance between the two templates. Given the template length $N = 2 \cdot r \cdot \theta$, we indicate by $F_1 = f_{1,1} \dots f_{1,N}$ and $F_2 = f_{2,1} \dots f_{2,N}$ the iris templates and $M_1 = m_{1,1} \dots m_{1,N}$ and $M_2 = m_{2,1} \dots m_{2,N}$ the corresponding noise masks. We assume that the value $m_i = 1$ in the mask vector indicates that the bit is affected by noise and must be excluded from the computation. Moreover, we indicate with $\overline{a} = 1 - a$ the negation of a feature bit a. The weighted Hamming Distance (HD) can be calculated as:

$$\begin{aligned} HD &= \frac{\|(F_1 \oplus F_2) \wedge (\overline{M}_1 \wedge \overline{M}_2)\|}{N - \|M_1 \vee M_2\|} \\ &= \frac{\sum_j^N \left[(f_{1,j} \oplus f_{2,j}) \wedge (\overline{m}_{1,i} \wedge \overline{m}_{2,i})\right]}{N - \sum_{j=1}^N m_{1,j} \vee m_{2,j}}. \end{aligned} \qquad (3)$$

2) Face recognition: Face templates can be generated by relying on the *eigenfaces* method proposed by Tuck and Pentland in [19] providing a set of facial characteristics that can be used to describe all the faces into the database (the *face-space*), as eigenvectors do in linear algebra. The template associated to a face Γ, is the projection of Γ on the face space $\Omega = [\omega_1 \dots \omega_k]$. Each ω_j describes the contribution of the corresponding eigenface, e_j, in representing the input image. In order to find the image that best matches with Γ, the algorithm looks for the projection vector Ω_j among all database images, that minimizes the Euclidean distance

$$ED = \|\Omega - \Omega_j\| = \sqrt{\sum_{i=0}^{k} (\Omega_i - \Omega_{j,i})^2}. \qquad (4)$$

3) Multi-biometric score level fusion: In this work we choose to use multimodal fusion (face and iris) at score level. This is motivated by the fact that, following the considerations in Section II-B, face and iris recognition protocols allow to easily compute the match score in the encrypted domain. Moreover, we underline that both iris and face can be acquired at the same time in real applications, for example by using a smartphone camera. We summarize our fusion system in Figure 1.

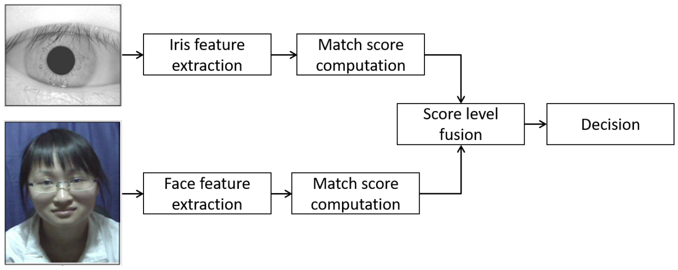

Fig. 1. General scheme of our fusion protocol.

Match scores generated by iris and face classifiers are characterised by a different range of values: the output of the iris protocol is a real number in $[0, 1]$, while the output of the face recognition protocol is a squared number in $[0, M], M \in \mathbb{R}$. Many ways have been proposed to overcome the problems generated by the differences between the scores provided by different biometric recognition systems (see [31] for more details). Among them, due to the characteristics of our MPC system, we choose a linear combination of the scores. Furthermore, to normalize the face matching score, we choose to use a min - max normalization method [31].

$$\text{face\_score\_norm} = \frac{\text{score} - \min_{face}}{\max_{face} - \min_{face}}, \qquad (5)$$

where $\min_{face}, \max_{face}$, indicate the minimum and maximum values of the face range. Since $min_{face} = 0$, if $i \in [0, 1]$ is the Hamming distance resulting from iris match, f the Euclidean distance between faces, and R the maximum value of the face recognition system, multimodal recognition corresponds to checking if the following inequality holds:

$$\alpha \cdot i + \beta \cdot \frac{f}{R} < t \qquad (6)$$

where $\alpha, \beta \in [0,1]$ are proper weights and $t \in [0,1]$ is the threshold.

The choice of the parameters α, β, T determines the trade-off between equal error rate (EER) and computational complexity (Section V-A). As in [34], [32] we choose $\beta = 1 - \alpha$ for the tests in the plain domain. More details are provided in Section IV.

IV. Proposed Protocol

In this section, we describe the details of SEMBA implementation. We start by presenting the security model (Section IV-A), then we present the implementation of the iris and face authentication protocols (Sections IV-B and IV-C) and the multimodal biometric protocol (Section IV-D). Finally, we discuss the security of the SPDZ protocols in Section IV-E.

A. *Security model*

All the protocols involve a client (the biometric owner) and a server that authenticates the identity of the client. Since in our computation we have only two parties, from now on, the SPDZ protocol is described for $n = 2$. On one hand the client does not want to reveal his biometric templates, on the other hand the server does not want to dislocate its records. Both client and server can be malicious, i.e. they may be interested in gaining as much information as possible on the other party even by deviating from the protocol. Considering that the SPDZ protocol involving n parties is secure up to $n-1$ malicious parties, we assume that only one between the client and the server can act maliciously. We underline that if both act maliciously, they obtain no real information about the counterpart. We also assume that the parties are connected through a secure channel providing privacy against eavesdroppers and any third party that can compromise the transmission.

B. *Iris authentication protocol in encrypted domain*

The SPDZ protocol supports operations modulo p. Each binary element of an iris feature is encrypted as a modulo p integer $\langle a \rangle$-share. For this reason, to implement the Hamming distance computation as in Equation (3), we must implement logical operations $\oplus, \vee, \wedge$ as a combination of integer operations $+, -, \cdot$. The correspondence between binary and integer operations is detailed in Table II.

TABLE II. Correspondence table between binary and integer operations

Binary	Integer
$a \oplus b$	$a + b - 2 \cdot a \cdot b$
$a \wedge b$	$a \cdot b$
$a \vee b$	$a + b - a \cdot b$
$\overline{a}$	$1 - a$

Let $F_1 = f_{1,1}, f_{1,2}, \cdots, f_{1,N}$ and $F_2 = f_{2,1}, f_{2,2}, \cdots, f_{2,N}$ be two binary iris feature templates, where N is the number of features, we indicate with $\langle F_i \rangle$ the vector containing shares of each element, i.e. the vector $\langle f_{i,1} \rangle, \langle f_{i,2} \rangle, \cdots \langle f_{i,n} \rangle$ for $i = 1, 2$.

Since $\overline{M}_1 \wedge \overline{M}_2$ is equivalent to $\overline{M_1 \vee M_2}$, the Hamming distance in Equation (3) can be computed as:

$$\frac{\sum_{i=1}^{N} \{(f_{1,i} + f_{2,i} - 2 \cdot f_{1,i} \cdot f_{2,i}) \cdot [1 - (m_{1,i} \vee m_{2,i})]\}}{N - \sum_{j=1}^{N} (m_{1,i} \vee m_{2,i})}, \tag{7}$$

where $m_{1,i} \vee m_{2,i} = m_{1,i} + m_{2,i} - m_{1,i} \cdot m_{2,i}$.

In SPDZ, as well as in other MPC protocols, division is a very expensive operation. Hence, instead of evaluating the division and then compare the distance with an acceptance threshold, the denominator is multiplied by the threshold before the comparison. By letting

$$num = \sum_{i=1}^{N} \{(f_{1,i} + f_{2,i} - 2 f_{1,i} f_{2,i}) \left[1 - (m_{1,i} \vee m_{2,i})\right]\} \tag{8}$$

and

$$den = N - \sum_{j=1}^{N} (m_{1,i} \vee m_{2,i}), \tag{9}$$

the authentication check corresponds to

$$num < t \cdot den. \tag{10}$$

As it can be seen from Equation (8), many share multiplications are needed to calculate the numerator, namely, N multiplications for each $F_1 \oplus F_2$, $M_1 \wedge M_2$ and $(F_1 \oplus F_2) \wedge (M_1 \wedge M_2)$. To compute the denominator, we can reuse the $m_{1,i} \vee m_{2,i}$ already computed for the numerator. Multiplication between shares requires data transmission, slowing down the computation. To optimize the protocol, we split the multiplication protocol into two parts. First we calculate $\langle \varepsilon_i \rangle = \langle f_{1,i} \rangle - \langle a \rangle$ and $\langle \delta_i \rangle = \langle f_{2,i} \rangle - \langle b \rangle$, for all $i = 1 \ldots N$. Then, to partially open the values, both server and client exchange shares by using a packet for all ε's and one for all δ's. In this way we need only two transmissions for N multiplications.

Complexity. Computing $F_1 \oplus F_2$ and $M_1 \wedge M_2$ requires N multiplications each, one for each element of the template; moreover, N multiplications are required to compute $(F_1 \oplus F_2) \wedge (M_1 \wedge M_2)$. The total cost associated to the computation of num is $3N$ multiplications but, as we explained above, we need only 6 transmissions. Computing den (Equation (9)) has a negligible complexity since $m_{1,i} + m_{2,i} - m_{1,i} \cdot m_{2,i}$ has already been calculated for all i in Equation (8). Moreover, we need a multiplication between den and t, and ℓ multiplications for the comparison, where ℓ is the number of bits necessary to represent a modulo p integer (see Section III-A1). Consequently, we need $3N + \ell + 1$ multiplications but only $2\ell + 7$ transmissions for the iris protocol.

C. *Face authentication in encrypted domain*

As for the iris, we assume that face features have already been computed according to the protocol described in Section III-B2, obtaining a set of k real features Ω_i that have been rounded to represent them in $\mathbb{F}_p$ (in the next we avoid the round operator for simplicity). Given the projection Ω of the query face image, the face-based biometric authentication protocol must evaluate the Euclidean distance ED as in Equation (4), and check if it is lower than a threshold t. Considering that the square root cannot be evaluated efficiently in SPDZ, we instead compare the Squared Euclidean distance (SED) against the squared threshold:

$$\sum_{i=1}^{k}(\Omega_i - \Omega_{j,i})^2 < t^2. \tag{11}$$

In equation (11), Ω_i indicates an element of the face Ω and $\Omega_{j,i}$ the i-th element of the projection Ω_j. Moreover, as we did for the Hamming distance, we separate the square computation into two parts, so we need only one transmission to calculate SED.

Complexity. The computation of the Squared Euclidean distance requires the evaluation of k squares that can be parallelised, hence only one transmission is necessary. For the comparison we need ℓ products, as in the iris authentication protocol. Considering that squares and products have similar complexity, the complexity of the protocol is given by $k + \ell$ products.

D. Fusion in encrypted domain

We now describe our solution to implement the fusion protocol in the encrypted domain. As outlined in Equation (6), we use a linear combination of the matching scores; to avoid performing divisions, we evaluate

$$\alpha \cdot num \cdot R + \beta \cdot \text{SED} \cdot den < T \cdot den \cdot R, \tag{12}$$

where num and den stand for the numerator and denominator of the iris Hamming distance, i in (6), while SED, R and T stand for squared Euclidean distance score, face maximum range, and threshold.

The SPDZ framework does not allow the use of non-integer numbers, so α, β and T are scaled and approximated to integers in the interval $[0, 10]$. We chose this interval because it is accurate enough to obtain the same results achieved in the plain domain, and the resulting bitlength is small enough to make it possibile to represent $\alpha \cdot num \cdot R + \beta \cdot \text{SED} \cdot den$ and $T \cdot den \cdot R$ in $\mathbb{Z}_p$.

Complexity The previous formula requires three multiplications and six transmissions that cannot be run in parallel. Moreover, it needs ℓ multiplications for the comparison. In total, the linear fusion requires $\ell+6$ multiplications and $2\ell+12$ transmissions, plus the multiplications necessary to compute the Hamming and the squared Euclidean distances. The total complexity of the full multimodal protocol is $3N + \ell + 6$ multiplications and k squares, while it requires only $2\ell + 19$ transmissions.

TABLE III. COMPLEXITY SUMMARY. WE UNDERLINE THAT TRANSMISSION NUMBER DEPENDS ONLY ON p'S BITLENGTH ℓ.

	Multiplication	Squares	Transmissions
Iris	$3N + \ell + 1$	0	$2\ell + 7$
Face	ℓ	k	$2\ell + 1$
Multimodal	$3N + \ell + 6$	k	$2\ell + 19$

E. Protocol security

Relying on SPDZ tool, our protocols is secure in the UC model if at least one of the two parties is honest. In fact, according to [2], our SPDZ-based protocol is secure against $n - 1$ malicious adversaries, where $n = 2$ in our two-party computation scenario. The offline phase does not depend on the functionality evaluated and its security demonstration against active adversaries in the UC model is provided in [2]. The security demonstration of the online protocol is provided in the following theorem.

Theorem 1. *The online SPDZ implementation of SEMBA is computationally secure against any static adversary corrupting at most* 1 *party if* p *is exponential in the security parameter.*

Proof: The proof follows the security demonstration of the online SPDZ protocol in [2]. We rely on the simulator $\mathcal{S}_{\texttt{ONLINE}}$ defined in [2] to work on top of the ideal multibiometric authentication functionality $\mathcal{F}_{\texttt{ONLINE}}$, such that the adversary cannot distinguish among the simulator using the real function $\mathcal{F}_{\texttt{ONLINE}}$ and the real SPDZ-based implementation using multiplication triples generated offline. Input values broadcasted by both the simulator and honest players are uniform and it is not possible to distinguish among them. During execution, interaction with player is performed only during multiplication and squaring where partial opening reveals uniform values for both honest parties and simulator. Also MACs have similar distribution in both the protocol and the simulation. If the protocol does not abort due to a cheat detection, both the real and the simulated runs output the decision bit. In the simulation the decision bit is obtained by a correct evaluation of the multi-biometric function on the inputs provided by the player. In real SPDZ-based implementation the adversary can cheat in the MAC check with probability $2/p$. Hence the probability that the adversary can distinguished the simulated environment from the real one is negligible if p is exponential. The adversary is not able to obtain the inputs of the honest player because if the protocol does not abort, he can observe only its input, the input shares received by the other party and the final result. To obtain the original inputs of the honest party, the adversary should be able to solve the inequality in (6), which has $N_i + N_f$ unknown variables for the server and $N_i + N_f + 3$ for the client, where N_i and N_f are the number of features used to represent iris and face respectively. ■

V. SYSTEM TUNING

In this section, we present the results of the tests performed on plain data. We will use such results to choose the best parameters to build an efficient protocol working in the encrypted domain. Tests have been carried out on the "CASIA-IrisV1" database for irises and "CASIA-FaceV5 part 1" database for faces, both collected by the Chinese Academy of Sciences' Institute of Automation (CASIA) [35], [36].

The CASIA-IrisV1 database for irises [35] contains 756 grey-scale eye images with 108 unique irises (or classes) and 7 images for each of them. As in [12], we used a subset of the database for the tests, retaining only those images in wich the algorithm has well separated iris region from sclera and pupil. The resulting database contains 625 eye images.

The CASIA-FaceV5 Databases for faces part 1 [36] contains 500 face images of 100 subjects. The face images are captured using Logitech USB camera in one session. All face images are 16 bit color BMP files and the image resolution is 640 × 480 pixels.

We implemented and tested our SPDZ-based iris and multibiometric protocols on a desktop equipped with 8GB RAM processor Intel Core i3 CPU 550 @ 3.20 GHz Quad-Core running Ubuntu 14.04 LTS (64 bit) operative system. We developed the test using C++ language with GMP free library for arbitrary precision arithmetic, operating on signed integers, rational numbers, and floating-point numbers.

To implement the SPDZ protocol we chose the 46 bit prime number $p = 67280421310721$, which is big enough to allow all the needed modular operations and comparisons, and guarantee the security of the protocol. Server and client run on the same computer, and we used a socket to simulate the transmission channel.

A. Parameter optimisation

We performed tests on plain data, running the authentication protocol on each single biometric and then by fusing eigenfaces and iriscodes.

a) Iris: For testing the iris authentication protocol, we have chosen a radial resolution r ranging from 4 to 20 and an angular resolution θ between 100 and 200 (Table IV). As said above, we tested the protocol on 625 eye images and each one has been compared with all the others. To perform the tests in the plain domain, we used the Matlab code provided by L. Masek (iris recognition source code [37]) as part of his work [12]. As we can see from Table IV, the best accuracy is achieved by letting the angular resolution be equal to 160 angles and radial resolution equal to 20 corresponding to an iris feature vector of length 6400.

To reduce the EER, Masek ([12]) shifts n times the iris templates keeping the lowest Hamming distance score. In SPDZ this operation is computationally very expensive, so we could not afford it.

TABLE IV. IRISCODE EER (%) WITHOUT SHIFTING, AS A FUNCTION OF DIFFERENT VALUES OF r AND θ.

r	Angular Resolution θ					
	100	120	140	160	180	200
4	8.19	6.43	4.37	3.34	3.31	3.10
6	6.88	5.05	3.01	2.45	2.71	3.01
8	6.13	4.42	2.69	2.19	2.58	4.36
10	6.31	4.03	2.64	2.44	2.54	3.92
12	5.96	4.10	2.56	2.14	2.59	3.71
14	5.71	3.85	2.54	2.17	2.58	3.27
16	5.49	3.79	2.32	2.13	2.51	3.31
18	5.71	3.61	2.46	2.31	2.41	3.18
20	5.77	3.74	2.20	2.08	2.41	3.13

b) Face: We have implemented the eigenface protocol by using the Open Source Computer Video (openCV) library[1] and Matlab. Face images are 640×480 pixel and we transformed them in 256 grey level images. The protocol has been tested on 500 images. We used the algorithm provided in the openCV library to build k eigenfaces with $k = 1 \dots 10$. Each image is thus represented by a projection vector of length k. Each projection element is a 16-bit integer and the squared Euclidean distance has been calculated by using Matlab. We observed that the use of more than 5 projections does not provide any significant improvement (see Table V).

TABLE V. EIGENFACE EER (%) VALUES, AS A FUNCTION OF THE NUMBER OF PROJECTIONS.

k	EER (%)	k	EER (%)
1	28.77	6	17.01
2	17.37	7	16.19
3	16.62	8	16.51
4	16.59	9	16.38
5	16.08	10	16.09

c) Multimodal: We have evaluated the efficiency of the fusion protocol in the plain domain, by fusing the outcomes of face and iris sub-algorithms through a Matlab implementation. From Table IV, we have chosen some relevant iris configurations, based on the achieved EER or number of features. First of all, to better compare with the best iris result, we chose $r = 20$ and $\theta = 160$ resulting in $N = 6400$, then for each θ, we looked for the best accuracy under 4%, and finally we chose those configurations with EER similar to the previous ones but less features. Moreover, we varied α in the interval $[0, 1]$ and the number of eigenfaces k from 1 to 10. Table VI summarizes our results, showing the EER for each N and k.

As shown in Table VI, the same accuracy of the 6400 stand alone iris protocol (2.08%) can be reached with many different multi-biometric configurations, e.g. by using $N = 3600$ and $k = 7$ or even by using only 1600 iris features and $k = 1$. For the tests in the encrypted domain, between the two configurations with the same accuracy, we chose the last one, since it has lower bandwidth and computational complexity (see Table III and Table VIII). We can also notice that keeping $N = 1600$, but using 2 features for face representation, we can lower both accuracy and complexity. Generally, by using two eigenfaces, the best possibile accuracy is provided with 5760 iris features, however the same performance are obtained also by the combination of 3600 iris features and three face features. For this reasons, we tested several configurations in the encrypted domain, as summarised in Table VII.

VI. COMPLEXITY OF THE SPDZ PROTOCOL

We evaluated the computational complexity of our implementation of the SPDZ protocol (Section IV), by using the parameters chosen in the previous section (see Table IV and Table VII). Execution times are heavily affected by the number of multiplications. As we said in Section IV-B, when possible, we performed a single transmission, by packing data. To calculate the execution time, reported in Table IX, we used the *clock* function, measuring the CPU time of the process.

In SEMBA the number of transmission rounds depends only on the bitlength ℓ of the prime number p and not on the feature configuration, as it can be seen from Table III. On the contrary, the amount of data transmitted by each party also depends on the number of features used in the protocol. In fact, the iris authentication protocol has a bandwidth of $(6N + 2\ell + 2) \cdot \ell$ bits, while the multimodal protocol bandwidth is $\ell \cdot (6N + k + 2\ell + 12)$ bits. Since the complexity of the iris protocol is much higher than that of the face-based authentication protocol, the overhead introduced by the multimodal biometric authentication is of few bytes, as it can be seen from Table VIII. For this reason, the communication complexity remains almost constant switching from the iris to the multimodal protocol.

[1] http://opencv.org/about.html

TABLE VI. EER OF THE MULTIMODAL BIOMETRIC AUTHENTICATION PROTOCOL. THE FIRST THREE COLUMNS SHOW IRIS'S PARAMETERS: FEATURE'S NUMBER (N), RADIAL RESOLUTION (r), AND ANGULAR RESOLUTION θ. FOURTH COLUMN REPRESENTS IRIS AUTHENTICATION SYSTEM'S EER (%). ALL THE OTHERS COLUMNS CONTAIN THE EER'S OBTAINED BY FUSING AN IRIS TEMPLATE OF LENGTH N AND A FACE TEMPLATE WITH $k \in \{1 \dots 10\}$ EIGENFACES. WE HIGHLIGHTED IN BOLD THE CONFIGURATIONS THAT WE HAVE SELECTED FOR THE TESTS UNDER ENCRYPTION.

Iris Parameters				Number of Eigenfaces (k)									
N	r	θ	iris	1	2	3	4	5	6	7	8	9	10
6400	20	160	2.08	1.17	**1.15**	1.02	1.25	1.25	1.24	1.31	1.37	1.4	1.41
5760	16	180	2.51	1.26	**0.98**	1.01	1.22	1.38	1.36	1.43	1.47	1.49	1.50
5600	20	140	2.20	1.20	1.08	1.19	1.18	1.34	1.28	1.36	1.38	1.39	1.40
4800	20	120	3.74	1.90	1.97	1.65	1.98	2.04	2.15	2.11	2.17	2.2	2.21
3840	12	160	2.14	1.84	1.52	1.50	1.45	1.63	1.74	1.76	1.77	1.78	1.78
3600	10	180	2.54	1.36	1.23	**0.97**	1.65	1.82	1.99	2.07	2.15	2.17	2.19
3360	12	140	2.56	1.51	1.32	1.38	1.31	1.56	1.61	1.58	1.63	1.67	1.69
2560	8	160	2.19	1.50	1.24	1.19	1.15	1.39	1.49	1.59	1.6	1.61	1.62
2400	6	200	3.01	2.01	1.83	1.89	2.13	2.37	2.56	2.66	2.7	2.72	2.73
2160	6	180	2.71	1.92	1.74	1.82	1.98	2.04	2.09	2.07	2.13	2.16	2.18
1920	6	160	2.45	1.47	1.42	1.43	1.57	1.66	1.95	1.92	1.95	1.96	1.97
1600	4	200	3.10	**2.01**	**1.87**	1.85	2.41	2.37	2.56	2.67	2.69	2.71	2.71
1280	4	160	3.34	2.29	1.89	2.22	2.26	2.51	2.80	2.80	2.92	2.98	3.01

TABLE VII. EQUAL ERROR RATE OF IRIS AND MULTIMODAL BIOMETRIC AUTHENTICATION PROTOCOLS FOR DIFFERENT SETTINGS; α, t RESPECTIVELY STAND FOR FUSION COEFFICIENT AND THRESHOLD.

Iris	EER		Face	Fusion parameters	
N	Iris (%)	Fusion (%)	k	α	t
1600	3.10	2.01	1	0.80	0.35
1600	3.10	1.87	2	0.55	0.25
3600	2.54	0.97	3	0.55	0.25
5760	2.51	0.98	2	0.80	0.35
6400	2.08	1.15	2	0.80	0.35

TABLE VIII. COMMUNICATION COMPLEXITY FOR THE IRIS AND MULTIMODAL PROTOCOLS. IT IS IMPORTANT TO NOTICE THAT ADDING FEW EIGENFACES INCERASES THE BANDWIDTH BY A FEW BYTES ONLY.

Iris	bandwidth (KB)			Face
N	iris	multimodal	overhead	k
1600	53.24	53.30	0.06	1
3600	119.16	119.23	0.07	3
5760	190.35	190.42	0.07	2
6400	211.44	211.51	0.07	2

The main goal of our work was exploiting multimodality to reduce complexity while maintaining the same accuracy of the iris-based protocol. Moreover, our analysis shows that the multimodal protocol can also be used to lower the EER without a significant loss in terms of complexity. In the following, we discuss both cases.

a) Improved efficiency: The running time of the stand alone iris authentication protocol ranges from 0.03s for 1600 bits, up to 0.12s for a 6400 bit-long template in the malicious setting (see Table IX), while Luo et al. protocol [16] with masks needs 2.5s for 9600 bits and 0.56s for 2048 bits in the semi-honest setting. Moreover from Table VII and Table IX, it is evident that SEMBA can provide the same accuracy of the best stand alone iris protocol, but with lower execution time and computational complexity. As a matter of fact, the best EER for the standalone iris protocol is 2.08% for 6400 features corresponding to 19246 multiplications (Table III) in 0.12s, while in the fusion configuration for 1600 iris features and 1 eigenface feature, we need only 8744 multiplications (see Table III, where we consider squares as multiplications) to obtain an EER equal to 2.01% in about 0.03 seconds. On the contrary, the number of required transmissions increases from $2\ell + 7$ to $2\ell + 19$ (Table III), but it depends only on the bit length of p.

b) Improved accuracy: As an alternative to improve the computational complexity, the use of two biometries instead of one can be exploited to achieve a higher accuracy, at the cost of a slight increase of complexity with respect to the iriscode protocol. In fact, as shown in Table III, complexity depends heavily on the number of iris features, however by adding two eigenfaces it is possible to decrease the EER rate, while the number of multiplications increases only from $3N + \ell + 1$ to $3N + \ell + 6 + k = 3N + \ell + 8$ (as usual we consider squaring to be equivalent to multiplication). More generally, when we move to multimodal authentication, the total CPU time slightly increases with respect to the unimodal iris protocol, but the EER always decreases; by adding one more eigenface ($k = 2$) to the 1600 iris feature configuration considered above, we can have a better EER (1.87%) with the same time complexity (0.03 seconds). For the case of 5760 bit long iris template, the EER passes from from 2.1% for the unimodal authentication to 0.98% for the bimodal case with $k = 2$ (Table VII). Finally, keeping 0.98% as target accuracy, we highlight that we can reduce N to 3600 at the cost of an additional feature in the face representation ($k = 3$). In this case, computational complexity goes from 36926 to 19596 multiplications and time complexity decreases from 0.109s to 0.05s (Table VIII and Table IX).

TABLE IX. IRIS PROTOCOL TIME IN SPDZ SYSTEM.

Iris	CPU time		Face
N	Iris (s)	multimodal (s)	k
1600	0.029s	0.030	1
1600		0.030	2
3600	0.048	0.049	3
5760	0.11s	0.109	2
6400	0.12s	0.120	2

VII. CONCLUSIONS

In this paper, we have proposed SEMBA, a multimodal authentication system based on the MPC approach SPDZ [2], [1] secure against a malicious party. We have shown that by using a multi-modal system it is possible to improve the efficiency of the recognition process in terms of number of multiplications and evaluation time, without any loss of accuracy. In the same way, it is also possible to improve accuracy at the cost of a negligible increase of complexity. As an additional contribution, we adapted the iris and face authentication protocols to work in the SPDZ setting. A further additional complexity reduction is achieved by resorting to packed transmission of encrypted data involved in the secure multiplication protocol. As future work, we plan to extend our approach to even more biometric traits, like fingerprints, behavioral biometric and many others. Another interesting research direction could be to look for different algorithms and more efficient fusion rules to merge the match scores. We are also interested to test our protocol on mobile devices, in order to measure the complexity of the whole protocol, including also multi-biometric acquisition and feature extraction.

References

[1] I. Damgård, V. Pastro, N. Smart, and S. Zakarias, "Multiparty computation from somewhat homomorphic encryption," in *Advances in Cryptology–CRYPTO 2012*. Springer, 2012, pp. 643–662.

[2] I. Damgård, M. Keller, E. Larraia, V. Pastro, P. Scholl, and N. P. Smart, "Practical covertly secure mpc for dishonest majority–or: Breaking the spdz limits," in *European Symposium on Research in Computer Security*. Springer, 2013, pp. 1–18.

[3] "Biometrics security and privacy protection," *IEEE Signal Processing Magazine*, vol. 32, no. 5, Sept 2015.

[4] J. Bringer, H. Chabanne, and A. Patey, "Privacy-preserving biometric identification using secure multiparty computation: An overview and recent trends," *IEEE Signal Processing Magazine*, vol. 30, no. 2, pp. 42–52, 2013.

[5] M. Barni, G. Droandi, and R. Lazzeretti, "Privacy protection in biometric-based recognition systems: A marriage between cryptography and signal processing," *IEEE Signal Processing Magazine*, vol. 32, no. 5, pp. 66–76, 2015.

[6] O. R. Michael, "How to exchange secrets by oblivious transfer," Technical Report TR-81, Aiken Computation Laboratory, Harvard University, Tech. Rep., 1981.

[7] A. C. Yao, "How to generate and exchange secrets," in *Annual IEEE Symposium on Foundations of Computer Science*, 1986, pp. 162–167.

[8] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in Cryptology - EUROCRYPT*. Springer, 1999, pp. 223–238.

[9] C. Gentry and S. Halevi, "Implementing Gentry's fully-homomorphic encryption scheme," *Advances in Cryptology–EUROCRYPT 2011*, pp. 129–148, 2011.

[10] O. Goldreich, "Secure multi-party computation," *Manuscript. Preliminary version*, pp. 86–97, 1998.

[11] A. Ross, K. Nandakumar, and A. K. Jain, "Introduction to multibiometrics," in *Handbook of Biometrics*. Springer, 2008, pp. 271–292.

[12] L. Masek, "Recognition of human iris patterns for biometric identification," *The University of Western Australia*, vol. 2, 2003.

[13] P. S. Pisa, M. Abdalla, and O. C. M. B. Duarte, "Somewhat homomorphic encryption scheme for arithmetic operations on large integers," in *Global Information Infrastructure and Networking Symposium (GIIS), 2012*. IEEE, 2012, pp. 1–8.

[14] Z. Erkin, M. Franz, J. Guajardo, S. Katzenbeisser, I. Lagendijk, and T. Toft, "Privacy-preserving face recognition," in *Privacy Enhancing Technologies*. Springer, 2009, pp. 235–253.

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

[16] Y. Luo, S.-c. S. Cheung, T. Pignata, R. Lazzeretti, and M. Barni, "An efficient protocol for private iris-code matching by means of garbled circuits," in *International Conference on Image Processing - ICIP*. IEEE, 2012, pp. 2653–2656.

[17] J. Bringer, M. Favre, H. Chabanne, and A. Patey, "Faster secure computation for biometric identification using filtering," in *IAPR International Conference on Biometrics - ICB*. IEEE, 2012, pp. 257–264.

[18] M. Blanton and P. Gasti, "Secure and efficient protocols for iris and fingerprint identification," in *European Symposium on Research in Computer Security - ESORICS*. Springer, 2011, pp. 190–209.

[19] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Conference on Computer Vision and Pattern Recognition - CVPR*. IEEE, 1991, pp. 586–591.

[20] J. Daugman, "How iris recognition works," *IEEE Transactions on circuits and systems for video technology*, vol. 14, no. 1, pp. 21–30, 2004.

[21] M. Kantarcioglu and O. Kardes, "Privacy-preserving data mining in the malicious model," *International Journal of Information and Computer Security*, vol. 2, no. 4, pp. 353–375, 2008.

[22] A. Abidin, "On privacy-preserving biometric authentication," in *International Conference on Information Security and Cryptology - Inscrypt*, 2016, pp. 169–186.

[23] M. A. Pathak and B. Raj, "Privacy-preserving speaker verification and identification using gaussian mixture models," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 2, pp. 397–406, 2013.

[24] D. Boneh, E.-J. Goh, and K. Nissim, "Evaluating 2-DNF formulas on ciphertexts," in *IACR Theory of Cryptography Conference - TCC*, vol. 3378. Springer, 2005, pp. 325–341.

[25] M. Kiraz and B. Schoenmakers, "A protocol issue for the malicious case of yao's garbled circuit construction," in *Symposium on Information Theory in the Benelux*, 2006, pp. 283–290.

[26] Y. Lindell, B. Pinkas, and N. P. Smart, "Implementing two-party computation efficiently with security against malicious adversaries," in *International Conference on Security and Cryptography for Networks*. Springer, 2008, pp. 2–20.

[27] B. Pinkas, T. Schneider, N. P. Smart, and S. C. Williams, "Secure two-party computation is practical." in *Asiacrypt*, vol. 9. Springer, 2009, pp. 250–267.

[28] J. B. Nielsen, P. S. Nordholt, C. Orlandi, and S. S. Burra, "A new approach to practical active-secure two-party computation." in *CRYPTO*, vol. 7417. Springer, 2012, pp. 681–700.

[29] Y. Lindell, "Fast cut-and-choose-based protocols for malicious and covert adversaries," *Journal of Cryptology*, vol. 29, no. 2, pp. 456–490, 2016.

[30] P. Gasti, J. Šeděnka, Q. Yang, G. Zhou, and K. S. Balagani, "Secure, fast, and energy-efficient outsourced authentication for smartphones," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 11, pp. 2556–2571, 2016.

[31] A. A. Ross, K. Nandakumar, and A. Jain, *Handbook of multibiometrics*. Springer Science & Business Media, 2006, vol. 6.

[32] M. Gomez-Barrero, E. Maiorana, J. Galbally, P. Campisi, and J. Fierrez, "Multi-biometric template protection based on homomorphic encryption," *Pattern Recognition*, vol. 67, pp. 149–163, 2017.

[33] T. Veugen, R. de Haan, R. Cramer, and F. Muller, "A framework for secure computations with two non-colluding servers and multiple clients, applied to recommendations," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 3, pp. 445–457, 2015.

[34] R. Connaughton, K. W. Bowyer, and P. J. Flynn, "Fusion of face and iris biometrics," in *Handbook of Iris Recognition*. Springer, 2013, pp. 219–237.

[35] C. A. of Sciences' Institute of Automation, "Casia-irisv1." [Online]. Available: http://biometrics.idealtest.org/

[36] C. A. of Sciences Institute of Automation, "Casia-facev5." [Online]. Available: http://biometrics.idealtest.org/

[37] P. K. Libor Masek, "Matlab source code for a biometric identification system based on iris patterns." The School of Computer Science and Software Engineering, The University of Western Australia., 2003.