IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 10 October 2022, accepted 26 October 2022, date of publication 1 November 2022, date of current version 8 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3218703

RESEARCH ARTICLE

Efficient Biometric Identification on the Cloud With Privacy Preservation Guarantee

LINLIN YANG[1], CHENGLIANG TIAN[1], GONGJING ZHANG[1], LEIBO LI[2], AND HUANLI WANG[3]

[1]College of Computer Science and Technology, Qingdao University, Qingdao, Shandong 266071, China
[2]Shandong Institute of Blockchain, Ji'nan, Shandong 250101, China
[3]School of Environmental and Municipal Engineering, Qingdao University of Technology, Qingdao, Shandong 266520, China

Corresponding author: Chengliang Tian (tcl0815@gmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61702294, in part by the Natural Science Foundation of Shandong Province under Grant ZR2022MF250, and in part by the Applied Basic Research Project of Qingdao City under Grant 17-1-1-10-jch.

ABSTRACT Benefited from its reliability and convenience, biometric identification has become one of the most popular authentication technologies. Due to the sensitivity of biometric data, various privacy-preserving biometric identification protocols have been proposed. However, the low computational efficiency or the security vulnerabilities of these protocols limit their wide deployment in practice. To further improve the efficiency and enhance the security, in this paper, we propose two new privacy-preserving biometric identification outsourcing protocols. One mainly utilizes the efficient Householder transformation and permutation technique to realize the high-efficiency intention under the known candidate attack model. The other initializes a novel random split technique and combines it with the invertible linear transformation to achieve a higher security requirement under the known-plaintext attack model. Also, we argue the security of our proposed two protocols with a strict theoretical analysis and, by comparing them with the prior existing works, comprehensively evaluate their efficiency.

INDEX TERMS Biometric identification, privacy-preserving, householder transformation, cloud computing.

I. INTRODUCTION

Biometric identification uses a person's unique physical traits such as their iris, voice, face, fingerprints or behavioral features-such as gait, voice, keystrokes to authenticate, verify and identify them. Compared to traditional authentication methods (such as passwords or smart cards), biometrics cannot be lost, forgotten or stolen. As Schneier has said: ''You are your key'' [18]. Therefore, biometrics are easy to use and empower a seamless, frictionless user experience. Currently, biometric identification is widely used in government agencies and private enterprises, as well as organizations requiring access control and employee identification such as electronic health [11], industrial internet of things [12], wireless sensor networks [15], assistive robots [16]. Noteworthily, as Research and Markets Ltd. reported, the use of contactless biometric systems has greatly increased due to the spread of COVID-19 [21].

Generally, the biometric identification involves a dataset of biometrics and a query. The data owner (DO) performs a search task over the dataset to match the query data. However, in the current big data era, the quantity of biometric data is growing in an exponential speed, which causes heavy storage and computational burdens on the DO. Therefore, outsourcing the storage and the identification task to a cloud server has become a popular computing diagram. Within this diagram, a resource-constrained DO can enjoy the abundant storage and computational resource on a pay-as-you-demand manner. Although cloud-assisted biometric identification has been shown a promising application foreground in practice, it also faces several serious security challenges. On the DO side, due to the sensitivity of biometrics, the leakage and abuse of these data may lead to inestimable loss of life and properties [17]. On the cloud side, for outside financial

The associate editor coordinating the review of this manuscript and approving it for publication was Turgay Celik.

incentives, it may be curious about DO's privacy information and deliberately reveal these data to malicious attackers. Therefore, a well-designed biometric identification outsourcing protocol should ensure the privacy of DO's sensitive data. Meanwhile, since DO needs to spend additional cost to protect the privacy information, the privacy-preservation approach must be efficient. That is, compared with achieving the identification task by the DO itself (without outsourcing), the designed outsourcing protocol also should ensure the DO to gain decent computational savings. Consequently, designing efficient and privacy-preserving biometric identification outsourcing protocols becomes a hot topic.

A. RELATED WORK

In the past few years, many different privacy-preserving biometric identification protocols [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [13], [14], [19], [20], [22], [29], [30], [31], [32], [33] have been proposed. However, some of the existing protocols are inefficient or vulnerable to known attacks.

In the secure two-party computation model, the database owner (DO) and the user (QU) privately execute a biometric recognition algorithm while keeping their biometric data information blind to each other. Within this two-party setting, many privacy-preserving methods [5], [6], [7], [8], [9] have been brought forward. Erkin et al. [7] presented the first powerful privacy-enhanced biometric face recognition system. Sadeghi et al. [8] pointed out that Erkin et al.'s protocol [7] required costly online communication and computationally expensive operations on homomorphically encrypted data and thus could not be widely deployed in practical applications. Also, they designed a new privacy-preserving face recognition protocol that combining homomorphic encryption and garbled circuits [27], which somewhat improved the communication and computation efficiency. Later, Osadchy et al. [6] noted that both of these two protocols were designed with the classical eigenface recognition algorithm [28] which was inaccurate in recognizing some images. Hence, they developed a SCiFI system based on a novel face recognition algorithm which was very robust under unseen conditions. However, Barni et al. [5] claimed that face images were known to be fairly weak biometric traits and put forward a privacy-preserving protocol for fingerprint-based authentication. Subsequently, Huang et al. [9] pointed out protocols [5], [6], [7] did not support the computation of global minimum and, to address this issue and reduce the computation and bandwidth costs, they developed a protocol that used a new backtracking technique and an improved privacy-preserving Euclidean-distance technique. Then, Lagendijk et al. [29] and successive works [30], [31], [32] proposed solutions based on cryptographic primitives such as homomorphic encryption techniques. However, homomorphic encryption-based protocols, although secure enough, are not suitable for the real world due to their heavy complexity. In addition, most of these protocols are not designed under a cloud environment and thus cannot be directly applied to the outsourcing model.

The closely related work to the theme of our paper is the design of efficient and privacy-preserving cloud-assisted biometric identification protocols. In this setting, the DO outsources the database and related query operations to a resource-abundant cloud server. Wong et al. [20] developed an efficient asymmetric scalar-product-preserving encryption (APSE) to construct secure ciphertext database and query operations. Later, Yuan and Yu [1] pointed out that the protocol [20] did not take into account the collusion between the cloud server and malicious users. Once this collusion happens, the encrypted biometric database could be compromised. To avoid collusion attacks, they presented a new outsourcing protocol that could resist known-plaintext attacks and allowed the cloud server to collude with attackers. Subsequently, Wang et al. [2] proved that the protocol [1] could be cracked by eliminating randomness or exploiting Euclidean distance results, and thus it did not achieve the claimed security. To amend this security flaw, Wang et al. [2] developed an improved protocol CloudBI-II. However it involved multiple large matrix multiplications and was inefficient in practice. Moreover, this protocol was also proven to be insecure in [13] and [14]. For the sake of higher level of security, Zhang et al. [22] put forward a new protocol PTBI-II with perturbation technique. In their protocol, the central limit theorem was used to construct disturbance terms, which made the sum of these terms obey normal distribution so as to eliminate the influence of these. However, the protocol needs to add enough disturbance terms to achieve high accuracy, which causes a lot of redundancy. Following Wang et al.'s step, Zhu et al. [10] also aimed to remedy the security weakness of Yuan and Yu's protocol [1]. Meanwhile, on the basis of the protocol [1], Hu et al. [19] proposed a privacy-preserving biometric identification outsourcing protocol with somewhat homomorphic encryption technique. Nevertheless, Liu et al. [3] pointed out that Zhu et al.'s and Hu et al.'s designs had inherent defects and could not resist known-plaintext attack. Thus, they presented a secure protocol by inserting the threshold value into the encryption algorithm. Unfortunately, Kim et al. [4] proposed a statistical-inference attack algorithm which could generate a false fingerprint vector passing through the identification process in protocol [3]. To sum up, designing a secure and efficient privacy-preserving biometric identification outsourcing protocol remains to be further studied.

B. OUR CONTRIBUTION

In this paper, we reinvestigate the study of privacy-preserving biometric identification outsourcing protocols and, to better balance the security and efficiency and provide more alternatives, we propose two new cloud-assisted biometric identification protocols. One mainly focuses on high efficiency with decent security model and the other seeks high security with admissible efficiency. Precisely, our main contribution can be summarized as follows.

- Firstly, we propose a high-efficiency privacy-preserving biometric identification outsourcing protocol. The

TABLE 1. Notations and their meanings.

Symbol	Description
n	the dimension of the biometric and query point
m	the size of the biometric database
$\boldsymbol{b}_i$	the ith biometric data point $\boldsymbol{b}_i = (b_{i1}, b_{i2}, \cdots, b_{in})^T$
$\boldsymbol{c_i}$	the ciphertext of the biometric data point $\boldsymbol{b}_i$
$\boldsymbol{B}$	the database $\boldsymbol{B} = \{(id_1, \boldsymbol{b}_1)\ (id_2, \boldsymbol{b}_2)\ \cdots\ (id_m, \boldsymbol{b}_m)\}$
$\boldsymbol{C}$	the encrypted biometric dataset $\boldsymbol{C} = \{(id_1, \boldsymbol{c}_1), (id_2, \boldsymbol{c}_2), \cdots, (id_m, \boldsymbol{c}_m)\}$
$\boldsymbol{q}_u$	the query point $\boldsymbol{q}_u = (q_{u1}, q_{u2}, \cdots, q_{un})^T$
$\boldsymbol{c}_{\boldsymbol{q}_u}$	the ciphertext of the query point $\boldsymbol{q}_u$
$\boldsymbol{H_v}$	the Householder matrix induced by a vector $\boldsymbol{v}$
$\boldsymbol{P}$	a permutation matrix

key technique ingredient underlying our design is the employment of the efficient Householder transformation, which can reduce the time-consuming matrix-matrix multiplication to efficient matrix-vector multiplication and make our first protocol highly efficient. Also, we argue the security of the protocol under the known candidate attack model.

- Secondly, we propose a high-security biometric identification outsourcing protocol. Through a newly raised random split technique and the invertible matrix transformation, we realize the privacy preservation of the dataset and the query under the KPA model. Also, the efficiency of this protocol is comparable to the prior works.
- Finally, we present a comprehensive evaluation on our two proposed outsourcing protocols by theoretically and experimentally comparing them with the prior existing works. Our theoretical and experimental analysis demonstrates that (1) both of our proposed protocols enable the DO to gain decent computational savings, (2) our protocol I is most efficient among these protocols and (3) our protocol II is more efficient in most of the stages than prior arts.

II. MODELS AND DESIGN GOALS

In this part, we will introduce the system model, threat model and our design goals. For ease of description, throughout the paper, we use lowercase bold letters to represent column vectors and use uppercase bold letters to denote matrices. The main symbols used in the paper are listed in the following Table 1.

A. SYSTEM MODEL

Our model is essentially the same as that of the aforementioned work [1], [2], [3], [4], [10], [13], [19], [20], [22], which involves three participants: the database owner (DO), the cloud server (CS), and the user (QU). As shown in FIGURE 1, DO owns a large-scale database $\boldsymbol{B} = \{(id_1, \boldsymbol{b}_1)\ (id_2, \boldsymbol{b}_2)\ \cdots\ (id_m, \boldsymbol{b}_m)\}$ where $\boldsymbol{b}_i$ refers to the index id_i's biometric data. To realize the secure database query operations with the assistance of the resource-abundant CS, DO sends the encrypted biometric data $\boldsymbol{C}$ to CS

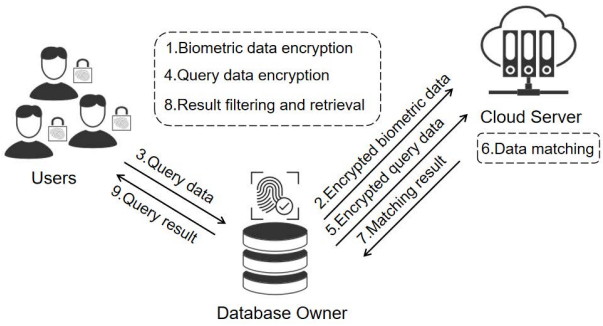

FIGURE 1. System model.

(*i.e.* steps 1 and 2 in the FIGURE 1). When QU issues a query request to DO, DO further encrypts the query data $\boldsymbol{q}$ and sends its ciphertext $\boldsymbol{q}'$ to CS (*i.e.* steps 3, 4 and 5 in the FIGURE 1). Then CS finds the best match with $\boldsymbol{q}'$ in $\boldsymbol{C}$ and returns the index of this match to DO (*i.e.* steps 6 and 7 in the FIGURE 1). Finally, DO finds the plaintext data corresponding to the index returned from CS, and checks the distance between this data and the query data with the similarity threshold. According to the calculation result, DO sends "True" or "False" to QU (*i.e.* steps 8 and 9 in the FIGURE 1).

Here, as illustrated in Yuan and Yu's protocol [1], we assume the data stored by DO and the query of QU are FingerCodes extracted from fingerprint images by using the Filterbank-based approach [23]. Usually, each FingerCode is an n-dimensional vector (n is generally equal to 640), and each entry is an 8-bit integer. The similarity of two FingerCodes $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)^T$ and $\boldsymbol{y} = (y_1, y_2, \cdots, y_n)^T$ is measured with Euclidean distance:

$$dist_{\boldsymbol{xy}} = \sqrt{\sum_{j=1}^{n}(x_j - y_j)^2}.$$

If the distance is less than a predefined threshold τ, the two FingerCodes can be considered from the same user and the match is successful.

B. THREAT MODEL

Following prior arts [1], [2], [3], [4], [10], [13], [19], [20], [22], in our system, we treat the CS as "honest but curious". That is, it will honestly conduct the specified computation task. However, for financial intention, it may try to spy the privacy information of other participants as much as possible. Based on this concern, we assume that the (internal or outside) attacker knows the encryption algorithm apart from the secret key, and, according to the attacker's ability and knowledge in practical applications, we consider the following three attack scenarios.

1) ATTACK SCENARIO 1

The attacker can only capture the encrypted database and the encrypted queries. This exactly is the well-known ciphertext-only attack (COA) model in cryptography [25].

2) ATTACK SCENARIO 2

In addition to the encrypted database and the encrypted queries, the attacker can also obtain some plaintext data points, but does not know the ciphertext corresponding to these plaintext data points. This is similar to the known candidate attack model in database literature [26].

3) ATTACK SCENARIO 3

Besides the ability in attack scenario 2, the attacker can obtain a subset of plaintext data points and their corresponding encrypted data points. In real world, this scenario could correspond to the scenario that the CS are curious and can obtain many QUs' query points. This is consistent with the known-plaintext attack (KPA) model in cryptography [25].

Obviously, the attacker in the scenario 3 is more powerful than that in scenarios 1 and 2. In other words, a scheme is secure under the attack scenario 3 is surely secure under the attack scenarios 1 and 2.

C. DESIGN GOALS

On basis of the above-mentioned system and threat models, we aim to design privacy-preserving and efficient biometric recognition protocols under the cloud environment. More precisely, our design should fulfill the following requirements.

- Correctness. This is a least requirement that the design should enable the DO to correctly identify the QU's biometric data if the CS perform the assigned computational task honestly and without outside attacker.
- Privacy preservation. In the designed protocol, the DO's biometric database and QU's query request should be kept privacy from the CS and the attacker under some decent threat model.
- High-efficiency. The computation overhead on the DO side should be substantially smaller than that of performing the identification task by the DO itself (*i.e.*, without the assistance of the CS).

III. PRELIMINARIES

Before presenting the details of our designs, some relevant background knowledge and mathematical tools are necessary.

A. HOUSEHOLDER TRANSFORMATION

Householder transformation, also known as the elementary reflection, is a linear transformation first proposed by A.C Aitken in 1932 [24]. It transforms a vector into the mirror image of a hyperplane reflection, out of which, the transformation matrix is called Householder matrix, and the normal vector of the hyperplane is called Householder vector. Formally, given a Householder vector $\boldsymbol{m} \in \mathbb{R}^n$, the Householder transformation induced by $\boldsymbol{m}$ is a mapping over Euclidean space $\mathbb{R}^n$ defined as below

$$\mathcal{H}_{\boldsymbol{m}}(\boldsymbol{x}) = \boldsymbol{H}_{\boldsymbol{m}}\boldsymbol{x}, \forall \boldsymbol{x} \in \mathbb{R}^n,$$

where the Householder matrix $\boldsymbol{H}_{\boldsymbol{m}} = \boldsymbol{E} - 2\boldsymbol{m}\boldsymbol{m}^T/(\boldsymbol{m}^T\boldsymbol{m})$ and $\boldsymbol{E}$ denotes the identity matrix. the Householder transformation schematic is shown in Figure 2.

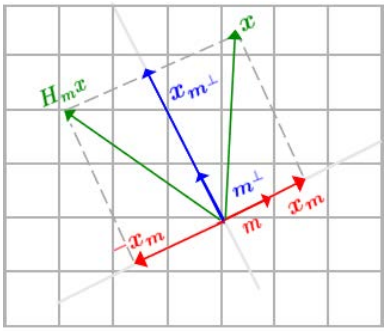

FIGURE 2. Householder transformation.

Geometrically, as illustrated in Figure 2, to obtain the mirror image $\boldsymbol{H}_{\boldsymbol{m}}\boldsymbol{x}$ of any vector $\boldsymbol{x}$ reflected on a hyperplane perpendicular to $\boldsymbol{m}$, we first consider the orthogonal projection of $\boldsymbol{x}$ onto $\boldsymbol{m}$, and get the orthogonal decomposition of $\boldsymbol{x}$:

$$\boldsymbol{x} = \boldsymbol{x}_{\boldsymbol{m}} + \boldsymbol{x}_{\boldsymbol{m}^\perp},$$

where $\boldsymbol{x}_{\boldsymbol{m}} = \frac{\langle \boldsymbol{x},\boldsymbol{m}\rangle}{\langle \boldsymbol{m},\boldsymbol{m}\rangle}\boldsymbol{m} = \frac{\boldsymbol{x}^T\boldsymbol{m}}{\boldsymbol{m}^T\boldsymbol{m}}\boldsymbol{m}$ and $\boldsymbol{x}_{\boldsymbol{m}^\perp} = \boldsymbol{x} - \boldsymbol{x}_{\boldsymbol{m}} = \boldsymbol{x} - \frac{\boldsymbol{x}^T\boldsymbol{m}}{\boldsymbol{m}^T\boldsymbol{m}}\boldsymbol{m}$. Then

$$\begin{aligned}\boldsymbol{H}_{\boldsymbol{m}}\boldsymbol{x} &= \left(\boldsymbol{E} - 2\frac{\boldsymbol{m}\boldsymbol{m}^T}{\boldsymbol{m}^T\boldsymbol{m}}\right)\boldsymbol{x} = \boldsymbol{x} - 2\frac{\boldsymbol{m}^T\boldsymbol{x}}{\boldsymbol{m}^T\boldsymbol{m}}\boldsymbol{m} \\ &= \boldsymbol{x} - 2\boldsymbol{x}_{\boldsymbol{m}} = -\boldsymbol{x}_{\boldsymbol{m}} + \boldsymbol{x}_{\boldsymbol{m}^\perp}.\end{aligned}$$

Now, we present a simple property of Householder transfromation.

Lemma 1: Let $\boldsymbol{H}_{\boldsymbol{m}}$ be a Householder matrix induced by an n-dimensional vector $\boldsymbol{m}$, Then, $\boldsymbol{H}_{\boldsymbol{m}}^T = \boldsymbol{H}_{\boldsymbol{m}}$ and, for any $\boldsymbol{x}$, $\boldsymbol{y} \in \mathbb{R}^n$, we have

$$\|\boldsymbol{H}_{\boldsymbol{m}}\boldsymbol{x} - \boldsymbol{H}_{\boldsymbol{m}}\boldsymbol{y}\| = \|\boldsymbol{x} - \boldsymbol{y}\|.$$

Proof: Clearly, $\boldsymbol{H}_{\boldsymbol{m}}^T = \boldsymbol{H}_{\boldsymbol{m}}$. For any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$, since $\boldsymbol{H}_{\boldsymbol{m}}^T\boldsymbol{H}_{\boldsymbol{m}} = \boldsymbol{H}_{\boldsymbol{m}}\boldsymbol{H}_{\boldsymbol{m}}^T = \boldsymbol{H}_{\boldsymbol{m}}^2 = \boldsymbol{E}$, we have

$$\begin{aligned}\|\boldsymbol{H}_{\boldsymbol{m}}\boldsymbol{x} - \boldsymbol{H}_{\boldsymbol{m}}\boldsymbol{y}\|^2 &= \|\boldsymbol{H}_{\boldsymbol{m}}(\boldsymbol{x} - \boldsymbol{y})\|^2 \\ &= \langle \boldsymbol{H}_{\boldsymbol{m}}(\boldsymbol{x} - \boldsymbol{y}), \boldsymbol{H}_{\boldsymbol{m}}(\boldsymbol{x} - \boldsymbol{y})\rangle \\ &= (\boldsymbol{x} - \boldsymbol{y})^T\boldsymbol{H}_{\boldsymbol{m}}^T\boldsymbol{H}_{\boldsymbol{m}}(\boldsymbol{x} - \boldsymbol{y}) \\ &= (\boldsymbol{x} - \boldsymbol{y})^T(\boldsymbol{x} - \boldsymbol{y}) \\ &= \|\boldsymbol{x} - \boldsymbol{y}\|^2,\end{aligned}$$

which implies $\|\boldsymbol{H}_{\boldsymbol{m}}\boldsymbol{x} - \boldsymbol{H}_{\boldsymbol{m}}\boldsymbol{y}\| = \|\boldsymbol{x} - \boldsymbol{y}\|$. □

B. PERMUTATION MATRIX

Let π be an n-order permutation over the set $\{1, 2, \cdots, n\}$ which is usually denoted as

$$\begin{pmatrix} 1 & 2 & \cdots & n \\ \pi(1) & \pi(2) & \cdots & \pi(n) \end{pmatrix}.$$

The $n \times n$ permutation matrix $\boldsymbol{P}_\pi$ induced by π is

$$\boldsymbol{P}_\pi = \begin{pmatrix} \boldsymbol{e}_{\pi(1)} \\ \boldsymbol{e}_{\pi(2)} \\ \vdots \\ \boldsymbol{e}_{\pi(n)} \end{pmatrix}$$

where $\boldsymbol{e}_{\pi(i)} = (0 \cdots 0\ 1\ 0 \cdots 0)$ denotes the row vector that the entry located in the $\pi(i)$-th position is 1 and the rest are 0s. Clearly, left-multiplying a matrix by a permutation matrix, the $\pi(i)$-th row of the original matrix will be adapted to the i-th row of the new matrix. Right-multiplying a matrix by permutation matrix, the i-th column of the new matrix will be the $\pi^{-1}(i)$-th column of the original matrix. Also, a random permutation can be efficiently constructed with the famous Knuth-Durstenfeld Shuffle algorithm (Algorithm 1) and has the following trivial property.

Lemma 2: For any $n \times n$ permutation matrix $\boldsymbol{P}_\pi$, we have

$$\|\boldsymbol{P}_\pi \boldsymbol{x} - \boldsymbol{P}_\pi \boldsymbol{y}\| = \|\boldsymbol{x} - \boldsymbol{y}\|,\ \forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$$

IV. OUR OUTSOURCING DESIGNS

To achieve our design goals stated in Section II-C, the key obstacle standing in front of us is to come up with an efficient and secure encryption technique that can preserve the order of the distances. According to different threat scenarios, in this section, we design two protocols to resist the attacks in scenario 2 and scenario 3, respectively.

A. OUR PROTOCOL I

First, we design a secure protocol under the attack scenario 2. Detailedly, the workflow of our first protocol proceeds as follows.

- **Key generation stage.** In this stage, DO generates an $(n+2)$-order random permutation π with Algorithm 1, an $(n+2)$-dimensional random vector $\mathbf{v} = (v_1 \cdots v_n\ v_{n+1}\ v_{n+2})^T$ and two reals r_1, r_2, where each entry v_i, r_1, r_2 are all randomly and uniformly chosen with λ bits and λ denotes the maximum bit size of the entries in the biometric data.
- **Data encryption and upload stage.** In this stage, the DO first extends each data point $\boldsymbol{b}_i = (b_{i1}, \cdots, b_{in})^T$ to an $(n+2)$-dimensional point $\boldsymbol{b}'_i = (b_{i1}, \cdots, b_{in}, r_1, r_2)^T$. Then, DO permutes $\boldsymbol{b}'_i$ with $\boldsymbol{P}$, and subsequently applies the Householder transformation $\boldsymbol{H}_{\boldsymbol{v}}$ to the permuted vector. That is, DO calculates

$$\boldsymbol{c}_i = \boldsymbol{H}_{\boldsymbol{v}} \times \boldsymbol{P} \times \boldsymbol{b}'_i, \tag{1}$$

  where $\boldsymbol{H}_{\boldsymbol{v}} = \boldsymbol{E} - 2\boldsymbol{v}\boldsymbol{v}^T/(\boldsymbol{v}^T\boldsymbol{v})$. Finally, DO uploads the blinded database $\boldsymbol{C} = \{(id_1, \boldsymbol{c}_1), (id_2, \boldsymbol{c}_2), \cdots, (id_m, \boldsymbol{c}_m)\}$ to the cloud server CS. After that, to save the storage, the DO can delete the databases $\boldsymbol{B} = \{(id_1, \boldsymbol{b}_1), (id_2, \boldsymbol{b}_2), \cdots, (id_m, \boldsymbol{b}_m)\}$ and $\boldsymbol{C}$.
- **Query encryption stage.** When QU issues a query request $\boldsymbol{q}_u = (q_{u1}, q_{u2}, \cdots, q_{un})^T$ to DO, DO first extends $\boldsymbol{q}_u$ to an $(n+2)$-dimensional data point

$$\boldsymbol{q}'_u = \left(q_{u1}, q_{u2}, \cdots, q_{un}, s_1^{(\boldsymbol{q}_u)}, s_2^{(\boldsymbol{q}_u)}\right)^T,$$

  out of which, $s_1^{(\boldsymbol{q}_u)}$ and $s_2^{(\boldsymbol{q}_u)}$ are two secret numbers that are randomly and uniformly selected reals with κ bits and re-selected for different queries. Then DO calculates $\boldsymbol{c}_{\boldsymbol{q}_u} = \boldsymbol{H}_{\boldsymbol{v}} \times \boldsymbol{P} \times \boldsymbol{q}'_u$ and sends $\boldsymbol{c}_{\boldsymbol{q}_u}$ to CS.
- **Biometric data matching stage.** After receiving the encrypted query data point $\boldsymbol{c}_{\boldsymbol{q}_u}$, for $i = 1, \cdots, m$, the cloud CS calculates the distance $dist_{i,q_u} = \|\boldsymbol{c}_i - \boldsymbol{c}_{\boldsymbol{q}_u}\|^2$ and finds the minimum in the set $\{dist_{i,q_u} \mid 1 \le i \le m\}$. Let $dist_{k,q_u} = \min\{dist_{i,q_u} \mid 1 \le i \le m\}$. Then, CS returns $(id_k, dist_{k,q_u})$ to DO.
- **Result confirmation stage.** After receiving the result $(id_k, dist_{k,q_u})$, DO calculates $dist_{q_u} = dist_{k,q_u} - (r_1 - s_1^{(\boldsymbol{q}_u)})^2 - (r_2 - s_2^{(\boldsymbol{q}_u)})^2$. Then it compares this distance with the threshold τ. If $dist_{q_u} < \tau$, DO confirms the QU is the person with identification id_k.

Algorithm 1 $\boldsymbol{P}(\boldsymbol{B}, 1^\lambda)$

Input: A matrix $\boldsymbol{B} \in \mathbb{Z}^{n \times m}$ of rank m and a security parameter λ.

Output: A random permutation matrix $\boldsymbol{P} \in \mathbb{Z}^{n \times n}$

```
Set π=E_n(identical permutation)
for i = n to 2
    Set j to be a random integer with 1 ≤ j ≤ i
    Swap π[j] and π[i]
for i = 1 to n
    for j = 1 to n
        p_ij = δ_{π(i),j} (δ_{π(i),j} = 1 when π(i) = j, otherwise δ_{π(i),j} = 0).
Return P = (p_ij)_{1≤i,j≤n}
```

B. ANALYSIS OF PROTOCOL I

Now, we present a strict analysis on the correctness and the security of our protocol I.

Correctness analysis. According to the correctness requirement in our design goals, we need to prove that, if the QU is with the identification id_k, it will pass the confirmation. Mathematically, we need to prove $\|\boldsymbol{b}_k - \boldsymbol{q}_u\|^2 < \tau$. Equivalently, we need to explain that the value $dist_{q_u}$ calculated in our design is exactly $\|\boldsymbol{b}_k - \boldsymbol{q}_u\|^2$. In fact, in our design,

$$\begin{aligned} dist_{q_u} &= dist_{k,q_u} - (r_1 - s_1^{(\boldsymbol{q}_u)})^2 - (r_2 - s_2^{(\boldsymbol{q}_u)})^2 \\ &= \|\boldsymbol{c}_k - \boldsymbol{c}_{\boldsymbol{q}_u}\|^2 - (r_1 - s_1^{(\boldsymbol{q}_u)})^2 - (r_2 - s_2^{(\boldsymbol{q}_u)})^2 \\ &= \|\boldsymbol{H}_{\boldsymbol{v}} \times \boldsymbol{P} \times \boldsymbol{b_k}' - \boldsymbol{H}_{\boldsymbol{v}} \times \boldsymbol{P} \times \boldsymbol{q}'_u\|^2 - (r_1 - s_1^{(\boldsymbol{q}_u)})^2 \\ &\quad -(r_2 - s_2^{(\boldsymbol{q}_u)})^2. \end{aligned}$$

By Lemma 1 and Lemma 2, the above formula is

$$\begin{aligned} &= \|\boldsymbol{P} \times \boldsymbol{b_k}' - \boldsymbol{P} \times \boldsymbol{q}'_u\|^2 - (r_1 - s_1^{(\boldsymbol{q}_u)})^2 - (r_2 - s_2^{(\boldsymbol{q}_u)})^2 \\ &= \|\boldsymbol{b_k}' - \boldsymbol{q}'_u\|^2 - (r_1 - s_1^{(\boldsymbol{q}_u)})^2 - (r_2 - s_2^{(\boldsymbol{q}_u)})^2 \\ &= \left\| \begin{pmatrix} \boldsymbol{b_k} \\ r_1 \\ r_2 \end{pmatrix} - \begin{pmatrix} \boldsymbol{q}_u \\ s_1^{(\boldsymbol{q}_u)} \\ s_2^{(\boldsymbol{q}_u)} \end{pmatrix} \right\|^2 - (r_1 - s_1^{(\boldsymbol{q}_u)})^2 - (r_2 - s_2^{(\boldsymbol{q}_u)})^2 \\ &= \|\boldsymbol{b_k} - \boldsymbol{q}_u\|^2 + (r_1 - s_1^{(\boldsymbol{q}_u)})^2 + (r_2 - s_2^{(\boldsymbol{q}_u)})^2 \\ &\quad -(r_1 - s_1^{(\boldsymbol{q}_u)})^2 - (r_2 - s_2^{(\boldsymbol{q}_u)})^2 \\ &= \|\boldsymbol{b}_k - \boldsymbol{q}_u\|^2 \end{aligned}$$

Attack scenario 2. That is,

Theorem 1: Our protocol I is secure under the **Attack scenario** 2.

Proof: According to the settings of the **Attack scenario** 2, the attacker can obtain 1) ciphertexts $\boldsymbol{C} = \{(id_1, \boldsymbol{c}_1), (id_2, \boldsymbol{c}_2), \cdots, (id_m, \boldsymbol{c}_m)\}$, $\boldsymbol{c}_{\boldsymbol{q}_u}$ and $(id_k, dist_{k,q_u})$, and 2) some plaintexts in the biometric database whose corresponding ciphertexts are unknown. We know $\boldsymbol{c_i} = \boldsymbol{H_v} \times \boldsymbol{P} \times \boldsymbol{b}_i'$, $\boldsymbol{c}_{\boldsymbol{q}_u} = \boldsymbol{H_v} \times \boldsymbol{P} \times \boldsymbol{q}_u'$ and $dist_{k,q_u} = \|\boldsymbol{c_k} - \boldsymbol{c}_{\boldsymbol{q}_u}\|$. Precisely,

$$\boldsymbol{c_i} = \boldsymbol{H_v} \times \boldsymbol{P} \times \begin{pmatrix} \boldsymbol{b}_i \\ r_1 \\ r_2 \end{pmatrix}, \boldsymbol{c}_{\boldsymbol{q}_u} = \boldsymbol{H_v} \times \boldsymbol{P} \times \begin{pmatrix} \boldsymbol{q}_u \\ s_1^{(q_u)} \\ s_2^{(q_u)} \end{pmatrix}, \quad (2)$$

$$dist_{k,q_u} = \left\| \begin{pmatrix} \boldsymbol{b_k} \\ r_1 \\ r_2 \end{pmatrix} - \begin{pmatrix} \boldsymbol{q}_u \\ s_1^{(q_u)} \\ s_2^{(q_u)} \end{pmatrix} \right\|. \quad (3)$$

However, if $\boldsymbol{b}_i$ and $\boldsymbol{q}_u$ satisfy the above equation (2), then, at least for any $n \times n$ permutation matrix $\bar{\boldsymbol{P}}$, $\bar{\boldsymbol{P}}\boldsymbol{b}_i$ and $\bar{\boldsymbol{P}}\boldsymbol{q}_u$ also satisfy the equation (2). Consequently, the attacker can not distinguish the correct plaintext. □

Vulnerability Although our protocol I is secure under the **Attack scenario** 2, it can not resist the **Attack scenario** 3.

In fact, without loss of generality, assuming the attacker obtains ℓ plaintext-ciphertext pairs $\{\langle \boldsymbol{b_i}, \boldsymbol{c_i} \rangle \mid i = 1 \cdots, \ell\}$ with $\boldsymbol{b}_i = (b_{i1} \ \cdots \ b_{in})$ and $\boldsymbol{c_i} = (c_{i1} \ \cdots \ c_{in} \ c_{i(n+1)} \ c_{i(n+2)})$, then, by the encryption equation (1), the attacker can construct a system consisting of $\ell(n+2)$ equations:

$$\begin{cases} \pi(h_{11})b_{11} + \cdots + \pi(h_{1n})b_{1n} + \pi(h_{1(n+1)})r_1 \\ +\pi(h_{1(n+2)})r_2 = c_{11} \\ \pi(h_{21})b_{11} + \cdots + \pi(h_{2n})b_{1n} + \pi(h_{2(n+1)})r_1 \\ +\pi(h_{2(n+2)})r_2 = c_{12} \\ \quad \vdots \\ \pi(h_{(n+2)1})b_{11} + \cdots + \pi(h_{(n+2)n})b_{1n} + \pi(h_{(n+2)(n+1)})r_1 \\ +\pi(h_{(n+2)(n+2)})r_2 = c_{1(n+2)} \\ \quad \vdots \\ \pi(h_{11})b_{\ell 1} + \cdots + \pi(h_{1n})b_{\ell n} + \pi(h_{1(n+1)})r_1 \\ +\pi(h_{1(n+2)})r_2 = c_{\ell 1} \\ \pi(h_{21})b_{\ell 1} + \cdots + \pi(h_{2n})b_{\ell n} + \pi(h_{2(n+1)})r_1 \\ +\pi(h_{2(n+2)})r_2 = c_{\ell 2} \\ \quad \vdots \\ \pi(h_{(n+2)1})b_{\ell 1} + \cdots + \pi(h_{(n+2)n})b_{\ell n} + \pi(h_{(n+2)(n+1)})r_1 \\ +\pi(h_{(n+2)(n+2)})r_2 = c_{\ell(n+2)} \end{cases}, \quad (4)$$

where

$$\boldsymbol{H_v} = \begin{pmatrix} h_{11} & \cdots & h_{1(n+1)} & h_{1(n+2)} \\ \vdots & \cdots & \vdots & \vdots \\ h_{(n+2)1} & \cdots & h_{(n+2)(n+1)} & h_{(n+2)(n+2)} \end{pmatrix} \quad (5)$$

is the $(n+2) \times (n+2)$ Householder matrix, the permutation matrix $\boldsymbol{P}$ is induced by the permutation π and $\pi(h_{ij}) = h_{i\pi(j)}$. We can solve the above system with the linearized method which regards $\pi(h_{ij})$ as an unknown and treats $\pi(h_{i(n+1)})r_1$ and $\pi(h_{i(n+2)})r_2$ as two unknowns x_i and y_i respectively. Since $h_{ij} = h_{ji}$ and r_1 and r_2 are chosen same for each plaintext vector $\boldsymbol{b}_i$, the number of unknowns is $0.5n(n+1)+2n+2(n+2)$. As long as $\ell \geq (0.5n+4)(n+1)/(n+2)$, the attacker can recover $\boldsymbol{H_v} \times \boldsymbol{P}$, r_1 and r_2 by solving the linear system. Thus, the protocol I is broken.

Efficiency analysis. As an outsourcing design, our protocol should make the local client achieve decent computational savings. Now, we theoretically compare the DO's computation overhead of our protocol I with that of the algorithm without outsourcing.

Let $t_{KeyGen+DataEnc}$, t_{QuEnc}, t_{DataMa}, and t_{ReCon} denote the time overhead for the key generation and data encryption stage, the time overhead for the query encryption stage, the time overhead for the biometric data matching stage and the time overhead for the result confirmation stage, respectively. It should be noted that, in the data encryption and upload stage, the ciphertext data points can be efficiently calculated by the associativity of matrix multiplications. That is, for a biometric dataset $\boldsymbol{B}$, $\boldsymbol{H_vPB} = \boldsymbol{PB} - 2\boldsymbol{v}\boldsymbol{v}^T\boldsymbol{PB}/(\boldsymbol{v}^T\boldsymbol{v}) = \boldsymbol{PB} - 2\boldsymbol{v}(\boldsymbol{v}^T\boldsymbol{PB})/(\boldsymbol{v}^T\boldsymbol{v})$. DO only needs to execute a vector-matrix multiplication, which costs $\mathcal{O}(mn)$. That is $t_{KeyGen+DataEnc} = \mathcal{O}(mn)$. Also, this is a one-time work independent of the number of query requests. Similarly, due to the efficiency advantage of the Householder transformation in query encryption stage, the calculation of $\boldsymbol{c}_{\boldsymbol{q}_u}$ costs $t_{QuEnc} = \mathcal{O}(n)$. In the result confirmation stage, DO needs to calculate $dist_{q_u}$, which costs $t_{ReCon} = 2$ multiplications. Therefore, if the number of query requests is k, the DO's total overhead in our protocol I is $t_{total} = t_{KeyGen+DataEnc} + k(t_{QuEnc} + t_{ReCon}) = \mathcal{O}(mn) + \mathcal{O}(kn) = \mathcal{O}(mn + kn)$.

When DO performs the identification task by itself, it needs to perform mn multiplications for one query and the total computational overhead is $\mathcal{O}(kmn)$ for querying k times.

Overall, in case that the number of query requests $k > \mathcal{O}(\frac{mn}{mn-n})$, the time overhead of DO in our protocol I is much lower than that of DO performing the query task by itself.

C. OUR PROTOCOL II

To circumvent the above attack and seek a higher security under the **Attack scenario** 3, we further propose an improved privacy-enhanced protocol II. Through our analysis on the protocol I, the key flaw is that, in the encryption stage of protocol I, each biometric data point $\boldsymbol{b}_i$ is padded with the same random numbers r_1 and r_2, which incurs the number of unknowns is independent of the number ℓ of known plaintext-ciphertext pairs in system (4). Therefore, our new design must ensure that the number of unknowns always surpass the number of equations. That is, we should add more randomness in the encryption stage. Here, instead of encrypting the biometric data by padding random numbers, we split each $\boldsymbol{b}_i$ and $\boldsymbol{q}_u$ to two random vectors. More precisely, our protocol II specifies as follows.

- **Key generation stage.** In this stage, DO generates a random invertible matrix $\boldsymbol{M} = (m_{ij})_{1\le i,j\le n+1} \in \mathbb{R}^{(n+1)\times(n+1)}$ and its inverse $\boldsymbol{M}^{-1}$, and a random real vector $\boldsymbol{r} = (r_1\ r_2 \cdots\ r_m)^T$, where the entry m_{ij} $(1 \le i, j \le n+1)$ in the matrix $\boldsymbol{M}$ and $r_1, \cdots, r_m$ are all uniformly and randomly chosen with λ bits. Also, for each $1 \le i \le m$, DO constructs an $(n+1)$-dimensional point $\boldsymbol{b}_i^\star = (b_{i1}^\star, b_{i2}^\star, \cdots, b_{i(n+1)}^\star)^T$, where each entry $b_{ij}^\star$ is randomly and uniformly selected from $\{0, 1, -1\}$.
- **Data encryption and upload stage.** In this stage, DO extends each data point $\boldsymbol{b}_i = (b_{i1}, \cdots, b_{in})^T$ to an $(n+1)$-dimensional point $\hat{\boldsymbol{b}}_i = (b_{i1}, \cdots, b_{in}, -\frac{1}{2}\sum_{j=1}^{n} b_{ij}^2)^T$, and calculates $\boldsymbol{b}_i' = r_i \times \boldsymbol{b}_i^\star$ and $\boldsymbol{b}_i'' = \hat{\boldsymbol{b}}_i - \boldsymbol{b}_i'$. Finally, the ciphertext data point

$$\boldsymbol{c_i} = \boldsymbol{M}^T \times \boldsymbol{b}_i'' \tag{6}$$

  and the blinded database $\boldsymbol{C} = \{(id_1, \boldsymbol{c}_1), (id_2, \boldsymbol{c}_2), \cdots, (id_m, \boldsymbol{c}_m)\}$ is uploaded to the cloud CS.
- **Query encryption stage.** When QU issues a query request $\boldsymbol{q}_u = (q_{u1}, q_{u2}, \cdots, q_{un})^T$ to DO, DO first extends $\boldsymbol{q}_u$ to an $(n+1)$-dimensional data point $\hat{\boldsymbol{q}}_u = (q_{u1}, q_{u2}, \cdots, q_{un}, 1)^T$. Then, DO generates a random $(n+1)$-dimensional point $\boldsymbol{q}_u^\star = (q_{u1}^\star, q_{u2}^\star, \cdots, q_{u(n+1)}^\star)^T$, where $q_{uj}^\star$ is selected randomly at random from $\{0, 1, -1\}$. Also, DO calculates $\boldsymbol{q}_u'' = k^{(\boldsymbol{q}_u)}\boldsymbol{q}_u^\star$ and $\boldsymbol{q}_u' = \hat{\boldsymbol{q}}_u - \boldsymbol{q}_u''$, out of which, $k^{(\boldsymbol{q}_u)}$ is secret number that is randomly and uniformly selected reals with λ bits and re-selected for different queries. Finally, DO sends CS the encrypted query $\boldsymbol{c}_{\boldsymbol{q}_u} = \boldsymbol{M}^{-1} \times \boldsymbol{q}_u'$.
- **Biometric data matching stage.** After receiving $\boldsymbol{c}_{\boldsymbol{q}_u}$, for $i = 1, \cdots, m$, the cloud CS calculates the inner product $inpr_{i,q_u}' = \langle \boldsymbol{c_i}, \boldsymbol{c}_{\boldsymbol{q}_u} \rangle$ and returns the set $\mathcal{S} = \{(id_i, inpr_{i,q_u}') \mid 1 \le i \le m\}$ to DO.
- **Result confirmation stage.** After receiving the set $\mathcal{S}$, DO first updates the $inpr_{i,q_u}'$ with $inpr_{i,q_u} = inpr_{i,q_u}' + \langle \hat{\boldsymbol{b}}_i, \boldsymbol{q}_u'' \rangle + \langle \boldsymbol{b}_i', \boldsymbol{q}_u' \rangle$, and finds the maximum $inpr_{k,q_u} = \max\{inpr_{i,q_u} \mid 1 \le i \le m\}$. Then, it compares this distance with the threshold τ. If $inpr_{k,q_u} < \tau$, DO confirms the QU is the person with identification id_k.

D. ANALYSIS OF PROTOCOL II

Correctness analysis. For correctness, we need to explain why the maximum $dist_{k,q_u}$ captures the truth that $\boldsymbol{b}_k$ is the closest data point to the query $\boldsymbol{q}_u$. In fact, according to our design, for any $1 \le i \le m$,

$$\begin{aligned} inpr_{i,q_u} &= inpr_{i,q_u}' + \langle \hat{\boldsymbol{b}}_i, \boldsymbol{q}_u'' \rangle + \langle \boldsymbol{b}_i', \boldsymbol{q}_u' \rangle \\ &= \langle \boldsymbol{c}_i, \boldsymbol{c}_{\boldsymbol{q}_u} \rangle + \langle \hat{\boldsymbol{b}}_i, \boldsymbol{q}_u'' \rangle + \langle \boldsymbol{b}_i', \boldsymbol{q}_u' \rangle \\ &= \langle \boldsymbol{M}^T \times \boldsymbol{b}_i'', \boldsymbol{M}^{-1} \times \boldsymbol{q}_u' \rangle + \langle \hat{\boldsymbol{b}}_i, \boldsymbol{q}_u'' \rangle + \langle \boldsymbol{b}_i', \boldsymbol{q}_u' \rangle \\ &= \langle \boldsymbol{b}_i'', \boldsymbol{q}_u' \rangle + \langle \hat{\boldsymbol{b}}_i, \boldsymbol{q}_u'' \rangle + \langle \boldsymbol{b}_i', \boldsymbol{q}_u' \rangle \\ &= \langle \hat{\boldsymbol{b}}_i, \boldsymbol{q}_u' \rangle + \langle \hat{\boldsymbol{b}}_i, \boldsymbol{q}_u'' \rangle = \langle \hat{\boldsymbol{b}}_i, \hat{\boldsymbol{q}}_u \rangle \\ &= \sum_{j=1}^{n} b_{ij} q_{uj} - \frac{1}{2}\sum_{j=1}^{n} b_{ij}^2. \end{aligned}$$

Then, for two different data points $\boldsymbol{b}_{i_1}$ and $\boldsymbol{b}_{i_2}$ with $1 \le i_1, i_2 \le m, i_1 \ne i_2$,

$$\begin{aligned} inpr_{i_1,q_u} - inpr_{i_2,q_u} &= \sum_{j=1}^{n} b_{i_1 j} q_{uj} - \frac{1}{2}\sum_{j=1}^{n} b_{i_1 j}^2 \\ &\quad -(\sum_{j=1}^{n} b_{i_2 j} q_{uj} - \frac{1}{2}\sum_{j=1}^{n} b_{i_2 j}^2) \\ &= -\frac{1}{2}(\|\boldsymbol{b}_{i_1} - \boldsymbol{q}_u\|^2 - \|\boldsymbol{b}_{i_2} - \boldsymbol{q}_u\|^2). \end{aligned} \tag{7}$$

Clearly, by the above equation (7), if $inpr_{i_1,q_u} - inpr_{i_2,q_u} > 0$, then $\|\boldsymbol{b}_{i_1} - \boldsymbol{q}_u\|^2 < \|\boldsymbol{b}_{i_2} - \boldsymbol{q}_u\|^2$. In other words, the larger the value of $inpr_{i,q_u}$, the closer the Euclidean distance between $\boldsymbol{b}_i$ and $\boldsymbol{q}_u$. Consequently, the maximum $inpr_{k,q_u}$ means that $\boldsymbol{b}_k$ is the closest data point to the query $\boldsymbol{q}_u$.

Security analysis. Now, we argue the robust security of the protocol II under the **Attack scenario** 3. That is, we will prove

*Theorem 2: Our protocol II is secure under the **Attack scenario** 3.*

Proof: Similar with the vulnerability analysis of protocol I, in **Attack scenario** 3, the attacker can obtain ℓ linearly independent plaintext-ciphertext pairs, without loss of generality, $\{(\boldsymbol{b}_i, \boldsymbol{c_i}) \mid i = 1, \cdots, \ell\}$. By equation (6), $\boldsymbol{c_i} = \boldsymbol{M}^T \times \boldsymbol{b}_i'' = \boldsymbol{M}^T \times (\hat{\boldsymbol{b}}_i - r_i \times \boldsymbol{b}_i^\star)$. That is, the following system consisting of $\ell(n+1)$ equations can be derived:

$$\begin{cases} m_{11}(\hat{b}_{11} - r_1 b_{11}^\star) + \cdots + m_{1(n+1)}(\hat{b}_{1(n+1)} - r_1 b_{1(n+1)}^\star) \\ = c_{11} \\ m_{21}(\hat{b}_{11} - r_1 b_{11}^\star) + \cdots + m_{2(n+1)}(\hat{b}_{1(n+1)} - r_1 b_{1(n+1)}^\star) \\ = c_{12} \\ \qquad\qquad \vdots \\ m_{(n+1)1}(\hat{b}_{11} - r_1 b_{11}^\star) + \cdots + m_{(n+1)(n+1)} \\ (\hat{b}_{1(n+1)} - r_1 b_{1(n+1)}^\star) = c_{1(n+1)} \\ \qquad\qquad \vdots \\ m_{11}(\hat{b}_{\ell 1} - r_\ell b_{\ell 1}^\star) + \cdots + m_{1(n+1)}(\hat{b}_{\ell(n+1)} - r_\ell b_{\ell(n+1)}^\star) \\ = c_{\ell 1} \\ m_{21}(\hat{b}_{\ell 1} - r_\ell b_{\ell 1}^\star) + \cdots + m_{2(n+1)}(\hat{b}_{\ell(n+1)} - r_\ell b_{\ell(n+1)}^\star) \\ = c_{\ell 2} \\ \qquad\qquad \vdots \\ m_{(n+1)1}(\hat{b}_{\ell 1} - r_\ell b_{\ell 1}^\star) + \cdots + m_{(n+1)(n+1)} \\ (\hat{b}_{\ell(n+1)} - r_\ell b_{\ell(n+1)}^\star) = c_{\ell(n+1)} \end{cases} \tag{8}$$

Clearly, in above system, $c_{ij} (1 \le i \le \ell, 1 \le j \le n+1)$ and $\hat{b}_{ij} (1 \le i \le \ell, 1 \le j \le n+1)$ are known coefficients and $m_{ij} (1 \le i, j \le n+1)$, $r_i (1 \le i \le \ell)$ and $b_{ij}^\star (1 \le i \le \ell, 1 \le j \le n+1)$ are unknowns. With linearization technique, treating $r_k b_{kj}^\star m_{ij}, (1 \le k \le \ell, 1 \le i, j \le n+1)$ as an unknown, there are at least $\ell(n+1)^2$ unknowns in above

system. However, there are only $\ell(n+1)$ equations. In fact, for each new plaintext-ciphertext pair, $(n+1)$ new equations and at least $(n+1)^2$ new unknowns are introduced. The number of unknowns is always more than that of equations, and thus there exist at least exponentially many solution vectors. Thus, the probability that the attacker can uniquely determine the $\boldsymbol{M}$ is negligible.

For the case that the adversary can obtain different query data points and their corresponding ciphertext pairs $\{(\boldsymbol{q}_u^{(i)}, \boldsymbol{c}_{\boldsymbol{q}_u^{(i)}}) \mid i = 1, \cdots, \ell\}$, the analysis is essentially the same as above, we omit it here.

Finally, we consider the relative distance result $\{inpr'_{i,q_u} \mid i = 1, \cdots, m\}$. The adversary may attempt to derive some sensitive information such as $\boldsymbol{b}_i$ or $\boldsymbol{q}_u$ by exploring the relationship among the distances $inpr'_{i,q_u}(1 \le i \le m)$. In our protocol II,

$$\begin{aligned} inpr'_{i,q_u} &= \langle \boldsymbol{c_i}, \boldsymbol{c}_{\boldsymbol{q}_u} \rangle = \langle \boldsymbol{M}^T \times \boldsymbol{b}''_i, \boldsymbol{M}^{-1} \times \boldsymbol{q}'_u \rangle \\ &= \langle \boldsymbol{b}''_i, \boldsymbol{q}'_u \rangle \\ &= \langle \hat{\boldsymbol{b}}_i - r_i \times \boldsymbol{b}^\star_i, \hat{\boldsymbol{q}_u} - k^{(\boldsymbol{q}_u)} \times \boldsymbol{q}^\star_u \rangle \\ &= \langle \hat{\boldsymbol{b}}_i, \hat{\boldsymbol{q}}_u \rangle - r_i \langle \boldsymbol{b}^\star_i, \hat{\boldsymbol{q}_u} \rangle - k^{(\boldsymbol{q}_u)} \langle \boldsymbol{q}^\star_u, \hat{\boldsymbol{b}}_i \rangle \\ &\quad + r_i k^{(\boldsymbol{q}_u)} \langle \boldsymbol{b}^\star_i, \boldsymbol{q}^\star_u \rangle \end{aligned} \tag{9}$$

Suppose that the adversary can know $(\hat{\boldsymbol{b}}_i, inpr'_{i,q_u})$ for $i = 1, \cdots, \ell$ and the query $\hat{\boldsymbol{q}_u}$. The number of unknowns is far more than that of equations according to equation (9). Thus, there exist at least exponentially many solution vectors and the probability of the adversary recovering the correct $(r_i, k^{(\boldsymbol{q}_u)}, \boldsymbol{b}^\star_i, \boldsymbol{q}^\star_u)$ for $i = 1, \cdots, \ell$ is negligible. Even if the adversary can recover them, since the random number $(r_i, \boldsymbol{b}^\star_i,)$ (resp. $(k^{(\boldsymbol{q})}, \boldsymbol{q}^\star_u)$) are variant with different data point (resp. query point), the probability that the adversary can recover other $\boldsymbol{b}_i$s for $i \in \{\ell+1, \cdots, m\}$ (resp. other query point) is also negligible.

Overall, based on the above analysis, the attacker with the abilities described in **Attack scenario** 3 cannot break our protocol II. □

Efficiency analysis. Now, we argue that, comparing with performing the query task by the DO itself, our protocol II can make DO achieve decent computational savings as long as the query scale is relatively large.

We first analyze the time overhead of DO in our protocol II. In the key generation stage, the most time-consuming operation is the calculation of $\boldsymbol{M}^{-1}$, which costs $\mathcal{O}(n^3)$. In the data encryption and upload stage, since the encryption matrix is an invertible matrix $\boldsymbol{M}$ rather than a Householder matrix, it costs $\mathcal{O}(mn^2)$. Also, the above two stages are one-time work for DO. In the query encryption stage, it costs $\mathcal{O}(n^2)$. In the results confirmation stage, when DO updates the $inpr_{i,q_u}$, it needs to perform 2 multiplications and $\mathcal{O}(n)$ additions. Since $1 \le i \le m$, the total time overhead of this stage is $\mathcal{O}(m)$ multiplications. Therefore, when QU queries k times, the total time overhead of DO in our protocol II is $\mathcal{O}(n^3 + mn^2 + k(m+n^2))$

As presented earlier, the computational cost of DO performing one query task by itself is $\mathcal{O}(mn)$. Thus, for querying k times, the total computational overhead is $\mathcal{O}(kmn)$.

Overall, as long as the number of query requests $k > \mathcal{O}(\frac{n^3+mn^2}{mn-m-n^2})$, the time overhead of DO in our protocol II is much lower than that of DO performing the query task by itself.

V. THEORETICAL AND PRACTICAL PERFORMANCE COMPARISON AND EVALUATION

In this section, we will present a comprehensive efficiency evaluation on our proposed two protocols by comparing them with the existing protocols in both theory and practice.

A. THEORETICAL COMPARISON ANALYSIS

We compare the theoretical complexity of our protocols with four popular and representative protocols [1], [2], [3], [10], [22]. As introduced in Section IV-B, we use $t_{KeyGen+DataEnc}$, t_{QuEnc}, t_{DataMa}, and t_{ReCon} to denote the time overhead for the key generation and data encryption stage, the time overhead for the query encryption stage, the time overhead for the biometric data matching stage and the time overhead for the result confirmation stage, respectively. Table 2 compares the computational and communicational overhead of these protocols. As we can see from Table 2, our protocol I has lower computation and communication overheads than other protocols, which greatly saves the costs of DO and CS. Moreover, our protocols II seeks high security at cost of somewhat communicational overhead. However, the efficiency of the protocol II is still comparable with the protocol [3] of the same security level. To illustrate this point explicitly, we refine the comparison of the computational overhead between our protocol II and Liu et al.'s protocol [3] in Table 3. As seen from Table 3, our protocol II is more efficient in most of the stages and outperforms the protocol [3] in the total overhead. The following experimental evaluation part also confirms our theoretical analysis.

B. EXPERIMENTAL EVALUATION

To evaluate the practical performance of our protocols, we experimentally compares these privacy-preserving fingerprint identification protocols. We simulate the DO on a computer with an Intel Core i7-9700T 2.00GHz CPU and 16.0 GB RMA and set up the CS with 10 nodes, each with the same hardware configuration. Additionally, following the experimental settings in the protocols [1], [2], [3], [10], [13], [22], we use randomly generated 640-entry vectors to represent the fingerprint database.

1) COMPARISON WITH THE ALGORITHM WITHOUT OUTSOURCING

We firstly compare the DO's time cost in our protocol I and protocol II with that in the algorithm without outsourcing. Set the size $m = 1e5$ of the fingerprint database and let the number of queries change from 1 to 300. We depict the

TABLE 2. Comparison on the computational and communicational overheads of the existing representative protocols.

	Computational overhead				Communicational overhead			
Phases	$t_{KeyGen+DataEnc}$ (DO)	t_{QuEnc} (DO)	t_{ReCon} (DO)	t_{DataMa} (CS)	Database upload (DO→ CS)	Query upload (DO→ CS)	Result Comfirmation (DO→QU)	Result download (CS→DO)
Yuan et al.'s protocol [1]	$\mathcal{O}(mn^3)$	$\mathcal{O}(n^3)$	$\mathcal{O}(n)$	$\mathcal{O}(mn^2)$	$\mathcal{O}(mn^2)$	$\mathcal{O}(n^2)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
Wang et al.'s protocol [2]	$\mathcal{O}(mn^3)$	$\mathcal{O}(n^3)$	$\mathcal{O}(n)$	$\mathcal{O}(mn^3)$	$\mathcal{O}(mn^2)$	$\mathcal{O}(n^2)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
Zhang et al.'s protocol [22]	$\mathcal{O}(mn^3)$	$\mathcal{O}(n^2)$	$\mathcal{O}(n)$	$\mathcal{O}(mn)$	$\mathcal{O}(mn^2)$	$\mathcal{O}(n)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
Zhu et al.'s protocol [10]	$\mathcal{O}(mn^2+n^3)$	$\mathcal{O}(n^3)$	$\mathcal{O}(n)$	$\mathcal{O}(mn^2)$	$\mathcal{O}(mn)$	$\mathcal{O}(n^2)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
Liu et al.'s protocol [3]	$\mathcal{O}(mn^2+mn)$	$\mathcal{O}(n^2)$	$\mathcal{O}(1)$	$\mathcal{O}(mn)$	$\mathcal{O}(mn)$	$\mathcal{O}(n)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
Our proposed protocol I	$\mathcal{O}(mn)$	$\mathcal{O}(n)$	$\mathcal{O}(1)$	$\mathcal{O}(mn)$	$\mathcal{O}(mn)$	$\mathcal{O}(n)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
Our proposed protocol II	$\mathcal{O}(mn^2+n^3)$	$\mathcal{O}(n^2)$	$\mathcal{O}(m)$	$\mathcal{O}(mn)$	$\mathcal{O}(mn)$	$\mathcal{O}(n)$	$\mathcal{O}(1)$	$\mathcal{O}(m)$

TABLE 3. Refined comparison on the computational overheads between Liu et al.'s protocol and our proposed protocol II.

	Computational overhead			
Phases	$t_{KeyGen+DataEnc}$ (DO)	t_{QuEnc} (DO)	t_{ReCon} (DO)	t_{DataMa} (CS)
Liu et al.'s protocol [3]	$mn^2+12mn+27m+n^3$	$n^2+12n+26$	\	$mn+5m$
Our proposed protocol II	$mn^2+3mn+m+n^3$	n^2+2n+1	$2m$	$mn+m$

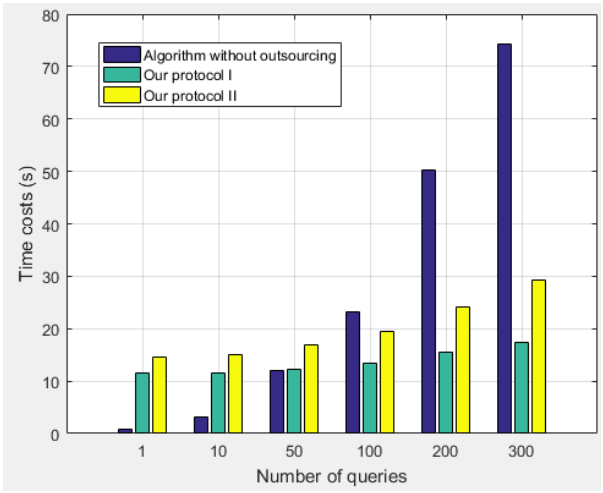

FIGURE 3. Comparison on the time costs of the DO.

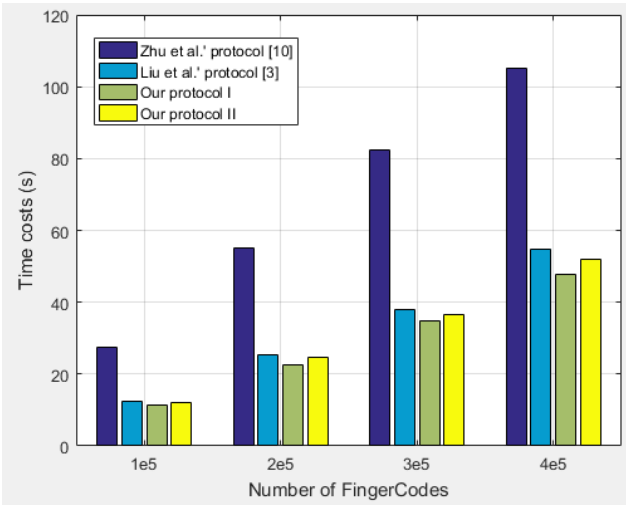

FIGURE 4. Comparison on the time costs of the key generation and data encryption stage.

experimental result in FIGURE 3. Clearly, without outsourcing, the DO's time cost increases linearly with the query size. While, in our protocols I and II, although the one-time work (*i.e.*, the *KeyGen* + *DataEnc* stage) is time-consuming, the DO's time cost is substantially less than the former if the number of the queries is relatively large. For the protocol I, the number is about larger than 50 and, for the protocol II, the number is about larger than 100. Furthermore, the more times the QU queries, the higher computational savings the DO can gain.

2) COMPARISON WITH THE EXISTING OUTSOURCING PROTOCOLS

In the following, we present an extensive comparison between our protocols with the protocols [3], [10] which have been shown to be experimentally more efficient than the other protocols [1], [2], [19].

a: KEY GENERATION AND DATA ENCRYPTION STAGE

FIGURE 4 shows the variance of the time cost of the key generation and data encryption stage for a single query with the number of FingerCodes changing from $1e5$ to $4e5$. As demonstrated in FIGURE 4, the time costs increase linearly for all the four protocols. The time cost of Zhu et al.'s protocol [10] is significantly higher than the other three protocols. Compared with protocol [10], our protocol I can save about 58% of the time cost and our protocol II can save about 55% of the time

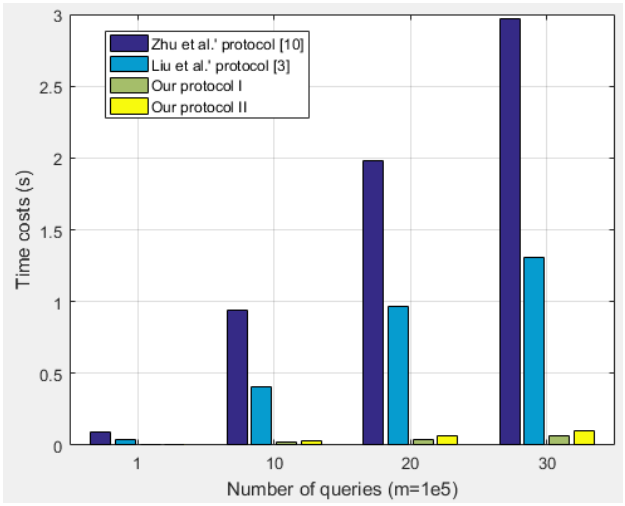

FIGURE 5. Comparison on the time costs of the query encryption stage.

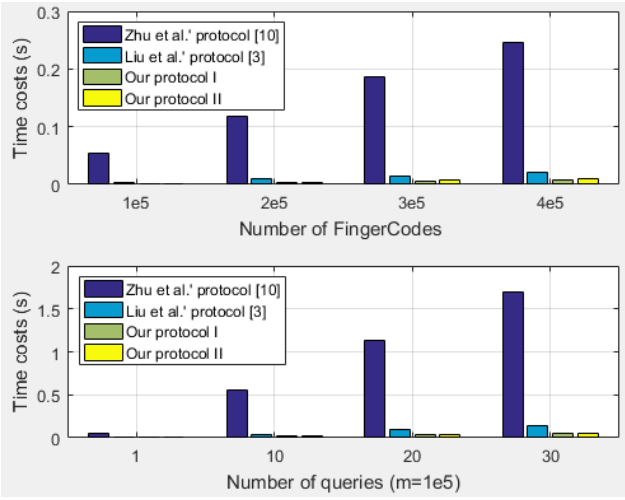

FIGURE 6. Comparison on the time costs of the data matching stage.

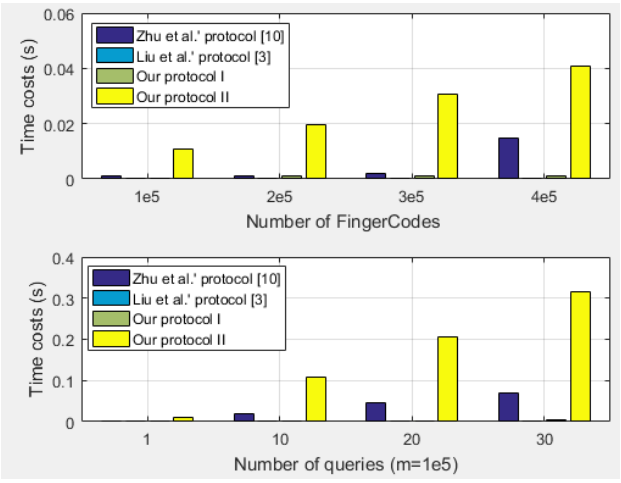

FIGURE 7. Comparison on the time costs of the result confirmation stage.

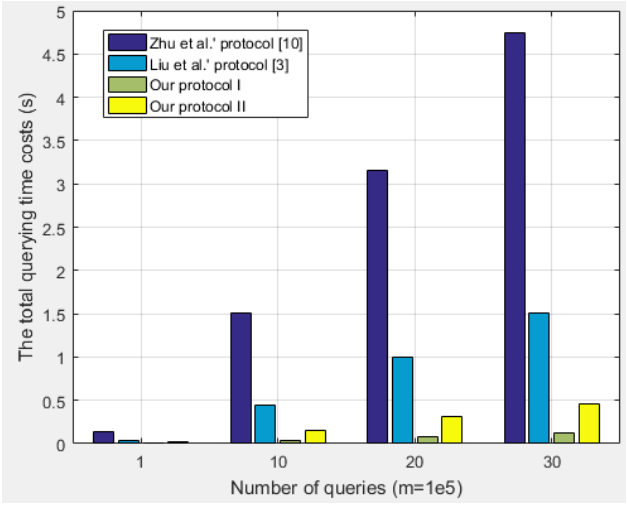

FIGURE 8. Comparison on the total querying time costs.

cost. The time cost of the protocol [3] is slightly higher than that of our protocol I as well as that of the protocol II. Our experimental results show that, compared with protocol [3], our protocol I can save about 10% of the time cost and our protocol II can save about 4% of the time cost.

b: QUERY ENCRYPTION STAGE

FIGURE 5 shows the time costs of the query encryption stage with the number of queries changing from 1 to 30 ($m = 1e5$). As shown in FIGURE 5, both of our proposed protocols are more efficient than the other two existing protocols. Also, the computational cost of our protocol I is the smallest, which is about 2% of that of Liu et al.'s protocol [3] and about 7% of that of Zhu et al.'s protocol [10].

c: BIOMETRIC DATA MATCHING STAGE

For this stage, we measure the variance of the time cost for a single query with the number of FingerCodes ranging from $1e5$ to $4e5$. Also, fixing the number of FingerCodes $m = 1e5$, we measure the time costs of this stage with the number of queries going from 1 to 30. The experimental results are visualized as FIGURE 6. We can draw the conclusion from FIGURE 6 that our protocol I and protocol II can reduce the computational cost on the CS side compared with protocols [3] and [10].

d: RESULT CONFIRMATION STAGE

For this stage, we measure the variance of the time cost for a single query with the number of FingerCodes ranging from $1e5$ to $4e5$. Moreover, fixing the number of FingerCodes $m = 1e5$, we measure the time costs of this stage as the number of queries goes from 1 to 30. FIGURE 7 depicts the experimental results, which shows that, compared with protocols [3] and [10], the time consumption of our protocols I is lower and the time consumption of protocol II is higher. The reason is that, in protocol II, this stage needs to update

the distance. However, even if the number of FingerCodes achieves $4e5$, the augmented time cost of our protocol II processing a query is less than $0.04s$ compared with the time costs of protocols [3], [10]. In the case that the number of queries achieves 30 (the number of FingerCodes is $1e5$), the augmented time consumption of our protocol II is less than $0.31s$ compared with the time costs of protocols [3], [10].

Finally, the total time cost for a single query which includes the time cost of query encryption stage, the time cost of biometric data matching stage and the time cost of result confirmation stage is an important index to evaluate the practical performance of an outsourcing protocol. To avoid the accidental errors, FIGURE 8 shows the variance of the total querying time cost with the number of queries ranging from 1 to 30 ($m = 1e5$). As seen from the FIGURE 8, our protocol II saves about 89% time cost compared with the protocol [10] and about 65% time cost compared with the protocol [3]. Specially, our protocol I's time cost is far less than those of other protocols. Therefore, our proposed protocols show remarkable performance in practice.

VI. CONCLUSION

In this paper, we design two biometric identification outsourcing protocols for different security and efficiency requirements. With rigorous theoretical argument and extensive experimental analysis, we comprehensively evaluate the security and efficiency of the proposed protocols. Our analysis indicates that our protocols can achieve decent security or efficiency advantages compared with the existing outsourcing protocols.

CONFLICTS OF INTERESTS

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this article.

REFERENCES

[1] J. Yuan and S. Yu, "Efficient privacy-preserving biometric identification in cloud computing," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2013, pp. 2652–2660, doi: 10.1109/INFCOM.2013.6567073.

[2] Q. Wang, S. Hu, K. Ren, M. He, M. Du, and Z. Wang, "CloudBI: Practical privacy-preserving outsourcing of biometric identification in the cloud," in *Computer Security*, vol. 9327. Cham, Switzerland: Springer, 2015, pp. 186–205, doi: 10.1007/978-3-319-24177-7_10.

[3] C. Liu, X. Hu, Q. Zhang, J. Wei, and W. Liu, "An efficient biometric identification in cloud computing with enhanced privacy security," *IEEE Access*, vol. 7, pp. 105363–105375, 2019, doi: 10.1109/ACCESS.2019.2931881.

[4] D. Kim and K. S. Kim, "A statistical inference attack on privacy-preserving biometric identification scheme," *IEEE Access*, vol. 9, pp. 37378–37385, 2021, doi: 10.1109/ACCESS.2021.3063693.

[5] M. Barni, F. Scotti, A. Piva, T. Bianchi, D. Catalano, M. Di Raimondo, R. D. Labati, P. Failla, D. Fiore, R. Lazzeretti, and V. Piuri, "Privacy-preserving fingercode authentication," in *Proc. 12th ACM Workshop Multimedia Secur.*, New York, NY, USA, 2010, pp. 231–240, doi: 10.1145/1854229.1854270.

[6] M. Osadchy, B. Pinkas, A. Jarrous, and B. Moskovich, "SCiFI—A system for secure face identification," in *Proc. IEEE Symp. Secur. Privacy*, May 2010, pp. 239–254, doi: 10.1109/SP.2010.39.

[7] Z. Erkin, M. Franz, J. Guajardo, S. Katzenbeisser, I. Lagendijk, and T. Toft, "Privacy-preserving face recognition," in *Proc. 9th Int. Symp. Privacy Enhancing Technol.*, vol. 5672. Berlin, Germany: Springer-Verlag, 2009, pp. 235–253, doi: 10.1007/978-3-642-03168-7_14.

[8] AR. Sadeghi, T. Schneider, and I. Wehrenberg, "Efficient privacy-preserving face recognition," in *Proc. 12th Int. Conf. Inf. Secur. Cryptol.*, vol. 5984. Berlin, Germany: Springer-Verlag, 2009, pp. 229–244, doi: 10.1007/978-3-642-14423-3_16.

[9] Y. Huang, L. Malka, D. Evans, and J. Katz, "Efficient privacy-preserving biometric identification," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, San Diego, CA, USA, 2011, pp. 90–98.

[10] L. Zhu, C. Zhang, C. Xu, X. Liu, and C. Huang, "An efficient and privacy-preserving biometric identification scheme in cloud computing," *IEEE Access*, vol. 6, pp. 19025–19033, 2018, doi: 10.1109/ACCESS.2018.2819166.

[11] P. Aparna and P. V. V. Kishore, "Biometric-based efficient medical image watermarking in E-healthcare application," *IET Image Process.*, vol. 13, no. 3, pp. 421–428, 2019, doi: 10.1049/iet-ipr.2018.5288.

[12] A. Krašovec, G. Baldini, and V. Pejović, "Opposing data exploitation: Behaviour biometrics for privacy-preserving authentication in IoT environments," in *Proc. 16th Int. Conf. Availability, Rel. Secur.*, New York, NY, USA, Aug. 2021, pp. 1–7, doi: 10.1145/3465481.3470101.

[13] Y. Zhu, Z. Wang, and J. Wang, "Collusion-resisting secure nearest neighbor query over encrypted data in cloud, revisited," in *Proc. IEEE/ACM 24th Int. Symp. Quality Service*, Jun. 2016, pp. 1–6, doi: 10.1109/IWQoS.2016.7590443.

[14] S. Pan, S. Yan, and W. Zhu, "Security analysis on privacy-preserving cloud aided biometric identification schemes," in *Proc. Australas. Conf. Inf. Secur. Privacy*. Berlin, Germany: Springer-Verlag, 2016, pp. 446–453, doi: 10.1007/978-3-319-40367-0_29.

[15] A. K. Das, "A secure and effective biometric-based user authentication scheme for wireless sensor networks using smart card and fuzzy extractor," *Int. J. Commun. Syst.*, vol. 30, no. 1, p. e2933, Jan. 2017, doi: 10.1002/dac.2933.

[16] S.-F. Baltanas, J.-R. Ruiz-Sarmiento, and J. Gonzalez-Jimenez, "A face recognition system for assistive robots," in *Proc. 3rd Int. Conf. Appl. Intell. Syst.*, New York, NY, USA, Jan. 2020, pp. 1–6, doi: 10.1145/3378184.3378225.

[17] E. Pagnin and A. Mitrokotsa, "Privacy-preserving biometric authentication: Challenges and directions," *Secur. Commun. Netw.*, vol. 2017, pp. 1–9, Oct. 2017, doi: 10.1155/2017/7129505.

[18] B. Schneier, "Inside risks: The uses and abuses of biometrics," *Commun. ACM*, vol. 42, no. 8, p. 136, Aug. 1999, doi: 10.1145/310930.310988.

[19] S. Hu, M. Li, Q. Wang, S. S. M. Chow, and M. Du, "Outsourced biometric identification with privacy," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 10, pp. 2448–2463, Oct. 2018, doi: 10.1109/TIFS.2018.2819128.

[20] W. K. Wong, D. W.-L. Cheung, B. Kao, and N. Mamoulis, "Secure kNN computation on encrypted databases," in *Proc. ACM SIGMOD Int. Conf. Manage. data*, New York, NY, USA, Jun. 2009, pp. 139–152, doi: 10.1145/1559845.1559862.

[21] *Biometrics Market: Global Industry Trends, Share, Size, Growth, Opportunity and Forecast 2021–2026*. [Online]. Available: https://www.researchandmarkets.com/reports/5264024/biometrics-market-global-industry-trends-share

[22] C. Zhang, L. Zhu, and C. Xu, "PTBI: An efficient privacy-preserving biometric identification based on perturbed term in the cloud," *Inf. Sci.*, vols. 409–410, pp. 56–67, Oct. 2017, doi: 10.1016/j.ins.2017.05.006.

[23] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based fingerprint matching," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 846–859, May 2000, doi: 10.1109/83.841531.

[24] A. C. Aitken, "XII.—Further numerical studies in algebraic equations and matrices," *Proc. Roy. Soc. Edinburgh*, vol. 51, pp. 80–90, Jan. 1932, doi: 10.1017/S0370164600023026.

[25] H. Delfs and H. Knebl, *Introduction to Cryptography: Principles and Applications*, 2nd ed. Berlin, Germany: Springer, 2010.

[26] K. Liu, C. Giannella, and H. Kargupta, "An attacker's view of distance preserving maps for privacy preserving data mining," in *Proc. 10th Eur. conf. Princ. Pract. Knowl. Discovery Databases (PKDD)*. Berlin, Germany: Springer-Verlag, 2006, pp. 297–308.

[27] A. C.-C. Yao, "How to generate and exchange secrets," in *Proc. 27th Annu. Symp. Found. Comput. Sci. (SFCS)*, Oct. 1986, pp. 162–167, doi: 10.1109/SFCS.1986.25.

[28] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 1991, pp. 586–591, doi: 10.1109/CVPR.1991.139758.

[29] R. L. Lagendijk, Z. Erkin, and M. Barni, "Encrypted signal processing for privacy protection: Conveying the utility of homomorphic encryption and multiparty computation," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 82–105, Jan. 2013, doi: 10.1109/MSP.2012.2219653.
[30] M. Gomez-Barrero, E. Maiorana, J. Galbally, P. Campisi, and J. Fierrez, "Multi-biometric template protection based on homomorphic encryption," *Pattern Recognit.*, vol. 67, pp. 149–163, Jul. 2017, doi: 10.1016/j.patcog.2017.01.024.
[31] M. K. Morampudi, M. V. N. K. Prasad, M. Verma, and U. S. N. Raju, "Secure and verifiable iris authentication system using fully homomorphic encryption," *Comput. Electr. Eng.*, vol. 89, Jan. 2021, Art. no. 106924, doi: 10.1016/j.compeleceng.2020.106924.
[32] J. J. Engelsma, A. K. Jain, and V. N. Boddeti, "HERS: Homomorphically encrypted representation search," *IEEE Trans. Biometrics, Behav., Identity Sci.*, vol. 4, no. 3, pp. 349–360, Jul. 2022, doi: 10.1109/TBIOM.2021.3139866.
[33] V. Bansal and S. Garg, "A cancelable biometric identification scheme based on Bloom filter and format-preserving encryption," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 34, no. 8, pp. 5810–5821, Sep. 2022, doi: 10.1016/j.jksuci.2022.01.014.

LINLIN YANG received the B.E. degree in internet of things from the Qilu University of Technology, in 2020. She is currently pursuing the M.S. degree with the College of Computer Science and Technology, Qingdao University. Her research interests include cloud computing security, secure computation outsourcing, and cryptography.

CHENGLIANG TIAN received the B.S. and M.S. degrees in mathematics from Northwest University, Xi'an, China, in 2006 and 2009, respectively, and the Ph.D. degree in information security from Shandong University, Ji'nan, China, in 2013. He held a post-doctoral position with the State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing. He is an Associate Professor with the College of Computer Science and Technology, Qingdao University. His research interests include lattice-based cryptography and cloud computing security.

GONGJING ZHANG is working as an Associate Professor with the College of Computer Science and Technology, Qingdao University. His research interests include big data security and privacy preservation.

LEIBO LI received the M.S. and Ph.D. degrees in information security from Shandong University, Ji'nan, China, in 2010 and 2014, respectively. He is currently working in the Shandong Institute of Blockchain. His research interests include secure multi-party computation and symmetric ciphers.

HUANLI WANG received the doctor's degree from Donghua University, in 2015. She is currently an Assistant Professor with the Qingdao University of Technology. Her research interests include computational materials science, big data analysis, and its applications in environmental sciences.