IDFace: Face Template Protection for Efficient and Secure Identification

Sunpill Kim[1†] Seunghun Paik[1†] Chanwoo Hwang[1] Dongsoo Kim[1] Junbum Shin[2] Jae Hong Seo[1*]
[1]Department of Mathematics & Research Institute for Natural Sciences, Hanyang University
[2]CryptoLab Inc.
{ksp0352, whitesoonguh, aa5568, frds37, jaehongseo}@hanyang.ac.kr, junbum.shin@cryptolab.co.kr

Abstract

As face recognition systems (FRS) become more widely used, user privacy becomes more important. A key privacy issue in FRS is protecting the user's face template, as the characteristics of the user's face image can be recovered from the template. Although recent advances in cryptographic tools such as homomorphic encryption (HE) have provided opportunities for securing the FRS, HE cannot be used directly with FRS in an efficient plug-and-play manner. In particular, although HE is functionally complete for arbitrary programs, it is basically designed for algebraic operations on encrypted data of predetermined shape, such as a polynomial ring. Thus, a non-tailored combination of HE and the system can yield very inefficient performance, and many previous HE-based face template protection methods are hundreds of times slower than plain systems without protection. In this study, we propose IDFace*, a new HE-based secure and efficient face identification method with template protection.* IDFace *is designed on the basis of two novel techniques for efficient searching on a (homomorphically encrypted) biometric database with an angular metric. The first technique is a template representation transformation that sharply reduces the unit cost for the matching test. The second is a space-efficient encoding that reduces wasted space from the encryption algorithm, thus saving the number of operations on encrypted templates. Through experiments, we show that* IDFace *can identify a face template from among a database of 1M encrypted templates in 126ms, showing only 2× overhead compared to the identification over plaintexts.*

1. Introduction

Recent advancements of the metric learning on the hypersphere have greatly improved the performance of face recognition systems (FRS) [8, 16, 30, 38, 39, 52, 67, 70], enabling various real-world applications of them. However, as the advanced FRS extracts more discriminative characteristics from a user's face image to his/her face template, privacy concerns regarding the secure storage of these templates become intensified. The advances on FRSs have focused on improving performance without protecting users' face templates, and using them without appropriate template protection methods can cause severe privacy problems. For example, several studies [19, 35, 36, 41, 48, 60–62] showed that face images can be reconstructed from corresponding unprotected face templates, even without seeing the internal parameters of the target FRS. Even worse, the reconstructed faces can be exploited for further malicious purposes, *e.g.*, impersonating the identity of the recovered faces in other FRSs, including commercial ones [41, 68]. Hence, for real-world applications such as identification at the airport or entrance of a building, as well-documented in GDPR [65], secure storage of templates becomes one of the most important security goals.

Faces are a form of biometrics, and biometric template protection (BTP) has been researched for a long time in various types of biometrics, including irises [23, 45, 69], fingerprints [12, 32, 56, 64], and recently faces [7, 11, 20, 40, 44]. The security notions of BTP are well established and have three standard requirements according to ISO/IEC 24745 [29]: irreversibility, revocability, and unlinkability. For pursuing concrete and provable security, cryptographic tools such as cryptographic hash functions and homomorphic encryptions (HE) [22] are widely employed on BTP designs. Despite strong security guarantees from these cryptographic tools, simultaneously achieving an appropriate level of security without sacrificing performance is still challenging. One obstacle is that cryptographic tools are designed to deal with specific types of data, *e.g.*, bit-strings or polynomials. In particular, contrary to classical biometrics such as iris and fingerprint, whose templates are usually represented as bit-strings [15, 49], most FRSs deal with real-valued templates. This makes designing BTPs for faces more challenging than other classical biometrics. Moreover, for applying BTPs in closed-set face recognition scenarios where the user's images can be utilized for training or fine-

*Corresponding author
†These co-first authors contributed equally to this work

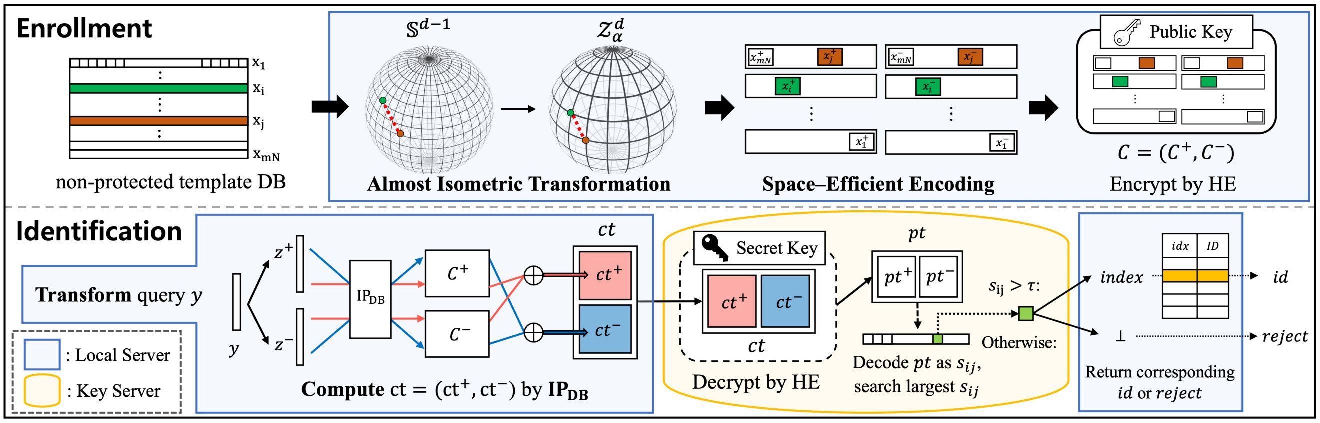

Figure 1. Overview of IDFace. Detailed explanations about each component will be presented in Section 4. Best viewed in color.

tuning the FRS, additional training for template protections is allowed to improve performance [14, 43, 57, 63]. However, this approach is not plausible in open-set face recognition scenarios where the train dataset is completely independent from users of a FRS.

FRSs are generally deployed in two different scenarios: verification and identification. While one identity is enrolled and verified in the verification scenario, multiple identities are enrolled and the closest one is identified in the identification scenario. In practice, rapid verification/identification procedures are usually more important than rapid enrollment. Contrary to the verification scenario, using heavy cryptographic tools for protection can be a burden in large-scale identification scenarios such as in airports, because their computational cost increases linearly with the number of identities. Recently, there have been several BTP proposals for identification [5, 6, 11, 17, 18, 20, 25, 34, 57] using HE, which enables arbitrary computation on encrypted data. However, despite several optimization techniques through SIMD (Single Instruction, Multiple Data) operations, the resulting performance of theirs remains impractical for large-scale applications.

1.1. Contribution

We find that the main cause of inefficiency in all previous approaches is that their use of HE was not tailored to coordinate with face templates. To overcome this limitation, we introduce two novel ideas to properly process templates so that they can be efficiently combined with HE. With these ideas, we propose IDFace, a *plug-and-play* HE-based face template protection method specialized for large-scale identification. Our IDFace achieves a drastic improvement in efficiency without significant accuracy degradation. In Fig. 1, we illustrated the brief overview of the proposed IDFace.

Our first idea is employing a novel transformation from a real-valued unit vector to a ternary vector comprised of $\{0, \pm 1\}$, while largely preserving the distance relationship. This facilitates harmonizing the aforementioned cryptographic tools with FRSs without significant accuracy degradation. In particular, our transformation enables computing the inner product from only additions. Since multiplications over encrypted data are much more expensive than additions, it massively accelerates the calculation of the inner product between templates when combined with HE.

Our second idea is introducing a space-efficient encoding for HE-based template protection schemes. We found that, for security reasons, each message slot of HE should occupy a considerably larger number of bits than that necessary for representing feature vectors. This gap becomes larger after applying the proposed transformation. To exploit the full capacity of each message slot, we propose a new encoding of feature templates to process much more of them simultaneously. This maximizes the advantage of SIMD operations while saving a large amount of storage.

We conducted extensive analyses on IDFace. First, we showed that our transformation does preserve the distance relationship to some extent under the suggested parameter settings. In addition, we experimentally verified that the accuracy drop by IDFace for various FRSs is less than 1% on famous benchmark datasets, including LFW [24], CFP-FP [59], AgeDB [55], and IJB-C [50]. The time spent for identification in IDFace is 126ms-753ms for a database with 1 million identities, depending on the amount of accuracy degradation, which only takes $\mathbf{2}\times$-$\mathbf{12}\times$ overhead compared to the setting without protection.

1.2. Related Works

We briefly review previous HE-based BTP proposals. In this paper, we will focus on HE-based BTPs only; surveys on other types of BTPs, *e.g.*, FC-based [33, 40, 42, 54, 63] or obfuscation-based [31, 53, 66], are deferred to Appendix A.

HE is a famous tool for privacy-preserving computation, and lots of HE-based BTPs have been proposed [5–7, 11, 17, 18, 20, 25, 34, 57]. The main advantage of utilizing HE is two-fold: since HE enables the computation of the matching score on an encrypted domain, this approach not only stores data and computes scores securely but also does not result in accuracy degradation. Moreover, thanks to SIMD operations, HE-based BTPs can leverage parallel computation in large-scale identification scenarios.

However, as [17] reported, a naive application of HE is quite far from the practical applications. Recent studies on designing HE-based BTPs have focused on optimization techniques to reduce the enormous computational cost of computing the matching score. [7] proposed a method to reduce the number of ciphertext rotations when computing the inner product. [6, 11, 34] focused on the waste of message slots in each ciphertext. [20] proposed a novel idea to encrypt the database for computing matching scores of a bunch of enrolled templates at the same time. Recently, a series of works [2–4] proposed a look-up table-based method during the score calculation, thus eliminating the need for homomorphic multiplications. In other directions, some proposals attempted to reduce the number of calculated matching scores [5, 18, 57] or propose an efficient matching algorithm [27] using scores. Many of them, including [6, 7, 20], considered dimensionality reduction for further speed-up on the identification process while sacrificing accuracy. Nevertheless, we found that all the previous approaches still suffer from expensive computational costs when a large number of identities, *e.g.* 1 million, are enrolled into the system. More detailed comparisons between theirs and our IDFace will be given in Tab. 2, Section 5.1.

2. Preliminaries

Notation. Throughout this paper, we will use the following notation. A vector is denoted by a lowercase letter shown in bold, such as $\mathbf{x} \in \mathbb{R}^d$, the components of which are denoted as a tuple $(x_1, x_2, \ldots, x_d)$. The standard inner product is denoted by $\langle \cdot, \cdot \rangle$. We denote $\mathbb{S}^{d-1}$ as a unit d-dimensional sphere embedded on $\mathbb{R}^d$. We denote a set of non-negative numbers less than $n \in \mathbb{N}$ as $[n]$.

2.1. Biometric Identification System

We first formalize the biometric identification system with a feature extractor, such as FRSs. For the set $\mathcal{I}$ of (preprocessed) biometrics and a metric space $(\mathcal{X}, d_{\mathcal{X}})$, the feature extractor is a function from $\mathcal{I}$ to $\mathcal{X}$. The output of the feature extractor is called (unprotected) biometric template. The feature extractor is expected to be a *similarity-preserving mapping* from the biometrics to the templates: two biometric templates from the same identity should be close enough, or vice versa. We remark that many recently proposed face recognition models [8, 16, 30, 38, 39, 52, 67, 70] adopt $\mathcal{X} = \mathbb{S}^{d-1}$ for $d = 512$ with angular metric as $d_{\mathcal{X}}$.

The operation of a biometric identification system consists of two phases, enrollment and identification, between two parties: the server and the user. In the enrollment, the user requests to enroll in the system by presenting his/her biometrics and an identifier. Then the server extracts a template from the user's biometrics through the feature extractor and stores it in the server's database with the corresponding identifier. In the identification, the user requests to identify him/herself by presenting the biometrics. The server first extracts a biometric template of the queried biometrics through the feature extractor and finds the identifier in the server's database whose corresponding template is closest to the extracted one. If the distance between these templates is under the predetermined threshold, then the server returns the found identifier; otherwise, the server returns "reject".

2.2. Homomorphic Encryption

HE is a cryptographic tool that enables executing arithmetic operations on an encrypted domain. Although there exist HE schemes supporting arbitrary arithmetic, called fully HE (FHE) [9, 10, 21], an additive homomorphic property suffices for our purpose. Our proposed method is not limited to a specific HE scheme; it supports both addition-only homomorphic encryption (AHE) schemes, such as the Paillier cryptosystem [58], and FHE schemes.

Many FHE schemes support SIMD additions for each message slot, which is denoted by $\oplus$. For any plaintexts $\mathbf{x}$ and $\mathbf{y}$ of dimension N, the following equality holds:

$$\mathsf{DEC}_{\mathsf{sk}}(\mathsf{ENC}_{\mathsf{pk}}(\mathbf{x} + \mathbf{y})) = \mathsf{DEC}_{\mathsf{sk}}(\mathsf{ENC}_{\mathsf{pk}}(\mathbf{x}) \oplus \mathsf{ENC}_{\mathsf{pk}}(\mathbf{y})),$$

where $\mathsf{ENC}_{\mathsf{pk}}$ and $\mathsf{DEC}_{\mathsf{sk}}$ are encryption and decryption algorithms of the FHE scheme, respectively.

Naturally, additions can be extended to scalar multiplications. For the sake of simplicity, we use a special notation $\odot$ for SIMD scalar multiplications on an encrypted domain satisfying the following relation for any scalar c:

$$\mathsf{DEC}_{\mathsf{sk}}(c \odot \mathsf{ENC}_{\mathsf{pk}}(\mathbf{x})) = \mathsf{DEC}_{\mathsf{sk}}(\mathsf{ENC}_{\mathsf{pk}}(c \cdot \mathbf{x})).$$

3. BTP from Database Encryption Scheme

Our approach for protecting face templates is basically to encrypt the database by AHE. We first formalize the notion of *database encryption* and give a generic construction of the biometric identification scheme from it.

3.1. Database Encryption Scheme

We begin with introducing the concept of *database encryption scheme*, whose required functionalities are to encrypt a bulk of templates simultaneously and calculate the matching score between the queried template and each enrolled template. The formal definition is given as follows:

Definition 1 (Database Encryption Scheme)**.** *The database encryption scheme is a pair of algorithms* ($\mathsf{ENC}_{\mathsf{DB}}$, $\mathsf{IP}_{\mathsf{DB}}$)*, each of whose functionality is as follows:*

- $\mathsf{ENC}_{\mathsf{DB}}$ *takes a bulk of biometric templates* $\mathbf{x}_1, \ldots, \mathbf{x}_N$ *and a public key* pk*, returning a ciphertext* $\mathcal{C}$*.*
- $\mathsf{IP}_{\mathsf{DB}}$ *takes a biometric template* $\mathbf{y}$ *and a ciphertext* $\mathcal{C}$*, returning a ciphertext of* $\langle \mathbf{x}_i, \mathbf{y} \rangle$ *for all plaintexts* $\mathbf{x}_i$ *of* $\mathcal{C}$*.*

Algorithm 1 $\mathsf{ENC}_{\mathtt{DB}}$ **(Base)**

Require: $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^T \in \mathbb{R}^{N \times d}$ and pk

Parse $\mathbf{X}$ as d column vectors $[\mathbf{v}_1, \ldots, \mathbf{v}_d]$

Compute $\mathbf{ct}_i \leftarrow \mathsf{ENC}_{\mathsf{pk}}(\mathbf{v}_i)$ for $\forall i \in [d]$

return $\mathcal{C} = (\mathbf{ct}_1, \ldots, \mathbf{ct}_d)$

Algorithm 2 $\mathsf{IP}_{\mathtt{DB}}$ **(Base)**

Require: $\mathbf{y} = (y_1, \ldots, y_d)$ and a ciphertext $\mathcal{C}$

Parse $\mathcal{C}$ as $(\mathbf{ct}_1, \ldots, \mathbf{ct}_d)$

$\mathbf{ct}_i = y_i \odot \mathbf{ct}_i$ for $\forall i \in [d]$

return $\mathbf{ct}_1 \oplus \cdots \oplus \mathbf{ct}_d$

We can view the construction of HERS [20], the previous state-of-the-art, as a database encryption scheme. The key idea of HERS was to view matrix-vector multiplication as a weighted sum of column vectors. If we denote $\mathbf{X} \in \mathbb{R}^{N \times d}$ as a matrix of which the row vectors are enrolled templates and $\mathbf{y} = (y_1, \ldots, y_d) \in \mathbb{R}^d$ as a vector requested to identify, the matching score between enrolled templates and $\mathbf{y}$ can be computed by a matrix-vector multiplication $\mathbf{X} \cdot \mathbf{y}$. If we denote $\mathbf{c}_i \in \mathbb{R}^N$ as the i'th column vector of $\mathbf{X}$ for $i \in [d]$, then we have that $\mathbf{X} \cdot \mathbf{y} = \sum_{i=1}^{d} y_i \cdot \mathbf{c}_i$. Hence, by encrypting each $\mathbf{c}_i$ during enrollment, we can compute the matching scores for the queried vector $\mathbf{y}_i$ in an encrypted domain via additions and scalar multiplications only.

Given an AHE scheme, we present a generic description of HERS as a database encryption $(\mathsf{ENC}_{\mathtt{DB}}, \mathsf{IP}_{\mathtt{DB}})$. $\mathsf{ENC}_{\mathtt{DB}}$ encrypts each N-dimensional column vector of $\mathbf{X}$ to an AHE-ciphertext $\mathbf{ct}_i$, and a final ciphertext is $\mathcal{C} = (\mathbf{ct}_1, \ldots, \mathbf{ct}_d)$. For given $\mathbf{ct}_i$'s and $\mathbf{y} = (y_1, \ldots, y_d)$, $\mathsf{IP}_{\mathtt{DB}}$ computes a ciphertexts of $(\langle \mathbf{x}_i, \mathbf{y} \rangle)_{i=1}^{N}$ by homomorphically computing a weighted sum of $\mathbf{ct}_i$'s with weights y_i's. The descriptions of both algorithms are given in Algorithm 1 and 2, respectively. We will consider $\mathsf{ENC}_{\mathtt{DB}}$ and $\mathsf{IP}_{\mathtt{DB}}$ as the baseline, and utilize them as subroutines in our method.

3.2. Secure Biometric Identification System from Database Encryption Scheme

We present a generic construction of a biometric identification system with template protection by a database encryption scheme. We first note that, to instantiate the database encryption scheme with an AHE, the secret key of the AHE should be kept secure. For this reason, we consider two servers for managing the encrypted database and the secret key separately: the *local server* ($\mathcal{S}_{local}$) that stores the public key pk and the encrypted database, and the *key server* ($\mathcal{S}_{key}$) that only stores the secret key sk. In this setting, $\mathcal{S}_{key}$ finds the identity of the largest score or returns *reject*, and $\mathcal{S}_{local}$ receives templates for enrollment and identification.

With the database encryption scheme ($\mathsf{ENC}_{\mathtt{DB}}$, $\mathsf{IP}_{\mathtt{DB}}$), $\mathcal{S}_{local}$ can encrypt a bunch of templates from enrollment by $\mathsf{ENC}_{\mathtt{DB}}$. In addition, by using $\mathsf{IP}_{\mathtt{DB}}$, $\mathcal{S}_{local}$ can compute matching scores between the queried template and stored ones in an encrypted domain. The identifier of the largest matching score can be found by $\mathcal{S}_{key}$ after decryption. From these idea, we construct the biometric identification system with two servers, $\mathcal{S}_{local}$ and $\mathcal{S}_{key}$, as follows.

In the enrollment, $\mathcal{S}_{local}$ is requested to enroll a set of biometric templates. It first separates the set of biometric templates into subsets of size N. For each subset $\mathbf{X}$ with the corresponding identities $\mathtt{ID}$, $\mathcal{S}_{local}$ runs $\mathsf{ENC}_{\mathtt{DB}}(\mathbf{X}, \mathsf{pk}) \rightarrow \mathcal{C}$, and adds $(\mathtt{ID}, \mathcal{C})$ into the bottom row of the database $\mathtt{DB}$.

In the identification, $\mathcal{S}_{local}$ receives a biometric template $\mathbf{y}$ and returns the corresponding identity, if there exists an enrolled template $\mathbf{x}$ such that $\langle \mathbf{x}, \mathbf{y} \rangle > \tau$ for a threshold τ. The precise process is as follows: assume that $\mathtt{DB}$ consists of $\{(\mathtt{ID}_i, \mathcal{C}_i)\}_{i=1}^{D}$. For each $i \in [D]$, $\mathcal{S}_{local}$ runs $\mathsf{IP}_{\mathtt{DB}}(\mathbf{y}, \mathcal{C}_i) \rightarrow \widehat{\mathcal{C}}_i$, whose output is the ciphertext containing inner products between $\mathbf{y}$ and plaintexts $\mathbf{x}$ of $\mathcal{C}_i$. Now, $\mathcal{S}_{key}$ obtains the decryption of $\{\widehat{\mathcal{C}}_i\}_{i=1}^{D}$ received by $\mathcal{S}_{local}$, which contains every matching score between the query and all enrolled identities in $\bigcup_{i=1}^{D} \mathtt{ID}_i$. Finally, $\mathcal{S}_{key}$ finds the index of the maximum matching score τ' and sends it to $\mathcal{S}_{local}$ if and only if $\tau' > \tau$, and $\mathcal{S}_{local}$ returns the corresponding id.

Later, we will follow this approach to construct the proposed BTP, IDFace, with an improved database encryption scheme from our novel techniques.

4. Proposed Method: IDFace

We now present two main ingredients of IDFace: almost isometric transformation that drastically improves computational efficiency with a small accuracy degradation, and space-efficient encoding for a bulk of templates that minimizes waste of plaintext space of HE. Using them, we construct our database encryption scheme and the proposed secure face identification system IDFace.

4.1. Almost Isometric Transformation

Face templates usually lie in the hypersphere $\mathbb{S}^{d-1}$ and the predominant operation in identification is taking the inner product between two unit vectors. We observed that an inner product value is relatively more influenced by components with large absolute values than those with small such values. Our main strategy is to transform the biometric template into a vector in $\{-1, 0, 1\}^d$ according to the magnitude of each component in the template[1]. More precisely, given a template, the transformation T_α with a parameter α first takes the top α components with larger absolute values. Then, those components are replaced with either 1 or -1 according to their signs, and the other components are set to 0. That is, the range of the transformation is a subset of $\{-1, 0, 1\}^d$, the vectors of which have only α nonzero components. We denote this set by $\mathcal{Z}_\alpha^d$.

[1] In fact, $\{-1, 0, 1\}^d$ is not a subspace of $\mathbb{S}^{d-1}$, so final normalization is required. Without loss of generality, we omit this because the number of nonzero components is fixed after the transformation.

With our transformation, taking an inner product can be done by a lookup operation, with addition and subtraction determined by the sign of each component. In particular, when combined with $\mathsf{IP}_{\mathsf{DB}}$, we can replace homomorphic scalar multiplications with look-ups, hence requiring much lower computational cost than the ordinary method.

Analysis on the Transformation. Although the computational advantage of our transformation is obvious, analyzing the extent to which it can preserve the distance relationship is not straightforward. To clarify this, we define a new notion of *almost isometry*, or $(\epsilon, \delta, \theta)$-isometry that catches the change of inner product values between two unit vectors of angle θ from the transformation. The formal definition of our almost isometry is provided in Definition 2.

Definition 2 ($(\epsilon, \delta, \theta)$-isometry)**.** *Let $T : \mathbb{S}^{d-1} \to \mathbb{S}^{d-1}$ be a function. Then for $\epsilon > 0, \theta \in (0, \pi)$, and $\delta \in [0, 1]$, T is called $(\epsilon, \delta, \theta)$-isometry if for $\mathbf{x}, \mathbf{x}' \in \mathbb{S}^{d-1}$ uniformly sampled from the set $\{(\mathbf{x}, \mathbf{x}') \in \mathbb{S}^{d-1} \times \mathbb{S}^{d-1} : \langle \mathbf{x}, \mathbf{x}' \rangle = \cos\theta\}$, $\Pr[|\langle T(\mathbf{x}), T(\mathbf{x}')\rangle - \langle \mathbf{x}, \mathbf{x}'\rangle| < \epsilon] > 1 - \delta$ holds.*

Here, the restriction $\langle \mathbf{x}, \mathbf{x}' \rangle = \cos\theta$ for sampling $\mathbf{x}$ and $\mathbf{x}'$ is necessary to ensure that the function preserves the inner product value even for two close vectors; note that the inner product value of two vectors uniformly sampled from $\mathbb{S}^{d-1}$ is close to 0 with an overwhelming probability. We also introduce a failure probability δ for relaxation.

We proved that the proposed transformation T_α satisfies the condition of $(\epsilon, \delta, \theta)$-isometry. However, due to space constraints, we only provide the result on the parameter ($\alpha = 341$) that minimizes ϵ for $d = 512$. The full statement, its proof, and further analyses on the choice of the transformation parameter are given in Appendix B and C.

Proposition 1 (Informal)**.** *For $d = 512$ and $\alpha = 341$, T_α is $(0.111, o(1), \theta)$-isometry, $\forall\, \theta \in (0, \pi)$.*

Remarks. Note that T_α is technically the same as the decoding algorithm of a real-valued error-correcting code by [40]. However, it plays a completely different role in our paper. The authors of [40] tried to facilitate the collision on $\mathcal{Z}_d^\alpha$ after the transformation, whereas we utilize it as an almost isometry. The almost-isometric property has neither been used nor even identified in their work.

We also note that the proposed transformation can be viewed as a ternary quantization on unit vectors. Several ternary quantization methods have been proposed for faster matrix multiplications on the inference of the neural network [26, 46, 47, 71]. Since a matrix multiplication can be viewed as a batched version of inner products, one can consider replacing our transformation to one of them. Nevertheless, we found that there are some desirable properties for adopting them to our IDFace. Due to space constraints, we provide discussions about them and our rationale for choosing the proposed transformation in Appendix E.

4.2. Space Efficient Encoding

The message space of AHEs is determined by the security parameter, regardless of practical applications. When employing them for face template protection, we found that a huge waste on the message space occurs if we do not carefully handle this issue. For example, Paillier cryptosystem [58] uses a 2048-bit message size for 112-bit security, and a typical parameter setting of CKKS [10] preserves a certain level of precision, *e.g.*. 50-bit, with 4096 message slots in 128-bit security. On the other hand, 16- or 32-bit precision is enough for representing face templates with an appropriate level of accuracy. Such a waste on the message space becomes worse after our transformation because the transformed template consists of only $-1, 0$ or 1.

We tackle this issue by further packing multiple templates into a single message slot. We first observe that the inner product value of two transformed templates by T_α ranges between $-\alpha$ and α since each transformed template has α nonzero components of ± 1. Handling both negative and positive values is rather cumbersome, and thus we separate a transformed template $\mathbf{x}$ into two positive binary vectors $\mathbf{x}^+$ and $\mathbf{x}^-$ such that $\mathbf{x} = \mathbf{x}^+ - \mathbf{x}^-$. Then, an inner product value of two transformed templates $\mathbf{x}$ and $\mathbf{y}$ is a sum of inner products between separations:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{x}^+, \mathbf{y}^+ \rangle + \langle \mathbf{x}^-, \mathbf{y}^- \rangle - \langle \mathbf{x}^+, \mathbf{y}^- \rangle - \langle \mathbf{x}^-, \mathbf{y}^+ \rangle.$$

Using separations instead of original templates, all the inner product values in the above lie between 0 and α. Now, we can encode m transformed templates $\mathbf{x}_1^\dagger, \ldots, \mathbf{x}_m^\dagger$ after separation into a single vector $\mathbf{x}^\dagger := \sum_{i=1}^m p^{i-1} \cdot \mathbf{x}_i^\dagger$ for $p > \alpha$ and $\dagger \in \{+, -\}$. Each $\mathbf{x}_i^\dagger$ can be decoded from $\mathbf{x}^\dagger$ by inductively calculating $\mathbf{x}_1^\dagger = \mathbf{x}^\dagger \pmod p$ and $\mathbf{x}_i^\dagger = (\mathbf{x}^\dagger - \sum_{j=1}^{i-1}(p^{j-1} \cdot \mathbf{x}_j^\dagger)) \cdot p^{1-i} \pmod p$ for $i = 2, \ldots, m$. If the message space of a message slot of AHE is n-bit, then we can encode at most $\lfloor \frac{n}{\log_2 \alpha} \rfloor$ templates at once. We call these two operations Encode and Decode, respectively.

We now show that Encode is compatible with $\mathsf{IP}_{\mathsf{DB}}$. Suppose that m transformed templates $\mathbf{x}_1, \ldots, \mathbf{x}_m$ are split into $(\mathbf{x}^+, \mathbf{x}^-)$ and each component is encrypted by $\mathsf{ENC}_{\mathsf{DB}}$, say $\mathcal{C}^+$ and $\mathcal{C}^-$. If a new template $\mathbf{y}$ is queried, we first split it by two vectors $\mathbf{y}^+$ and $\mathbf{y}^-$ and then run $\mathsf{IP}_{\mathsf{DB}}(\mathbf{y}^*, \mathcal{C}^\dagger)$ obtaining ciphertexts of $\langle \mathbf{x}^\dagger, \mathbf{y}^* \rangle$ for $\dagger, * \in \{+, -\}$. Since we appropriately select the bound p, the Decode with input $\langle \mathbf{x}^\dagger, \mathbf{y}^* \rangle$ returns $\langle \mathbf{x}_i^\dagger, \mathbf{y}^* \rangle$ for all $i \in [m]$. Finally, we can calculate the inner product values $\langle \mathbf{x}_i, \mathbf{y} \rangle$ for all $i \in [m]$.

Note that Encode is compatible with only one of the additions and subtractions. Thus, instead of directly computing $\langle \mathbf{x}, \mathbf{y} \rangle$, we only compute $\langle \mathbf{x}^\dagger, \mathbf{y}^* \rangle$ homomorphically for each $\dagger, * \in \{+, -\}$. We can reduce the number of decryptions by performing homomorphic additions to obtain two ciphertexts of $\langle \mathbf{x}^+, \mathbf{y}^+ \rangle + \langle \mathbf{x}^-, \mathbf{y}^- \rangle$ and $\langle \mathbf{x}^+, \mathbf{y}^- \rangle + \langle \mathbf{x}^-, \mathbf{y}^+ \rangle$, subtracting one from another after decoding.

4.3. Putting them All Together

With almost-isometric transformation and space-efficient encoding, we can improve the baseline database encryption scheme ($\mathsf{ENC}_{\mathsf{DB}}$, $\mathsf{IP}_{\mathsf{DB}}$). We denote each improved algorithm as IDFace.$\mathsf{ENC}_{\mathsf{DB}}$ and IDFace.$\mathsf{IP}_{\mathsf{DB}}$, respectively. We assume that the number of face templates is a multiple of mN, where both m and N are parameters selected by the underlying AHE scheme. These mN templates are represented by rows of a matrix $\mathbf{X} \in \mathbb{R}^{mN \times d}$. To process more templates, we can repeat the same procedure.

IDFace.$\mathsf{ENC}_{\mathsf{DB}}$ first applies our almost isometric transform T_α into each row vector of $\mathbf{X}$ and then encodes each m rows of $\mathbf{X}$ into two vectors $\mathbf{x}_i^+$ and $\mathbf{x}_i^-$ by our Encode algorithm for $i \in [N]$. Finally, it runs $\mathsf{ENC}_{\mathsf{DB}}$ for each $\mathbf{x}_i^+$ and $\mathbf{x}_i^-$ and returns a pair $\mathcal{C} = (\mathcal{C}^+, \mathcal{C}^-)$.

IDFace.$\mathsf{IP}_{\mathsf{DB}}$ takes a face template query $\mathbf{y} \in \mathbb{R}^d$ as input, and then applies T_β[2] to obtain a transformed template $\mathbf{z}$. Then, it splits $\mathbf{z}$ into positive part $\mathbf{z}^+$ and negative part $\mathbf{z}^-$. We parse the given $\mathcal{C}$ into ($\mathcal{C}^+$, $\mathcal{C}^-$) and compute $\mathbf{ct}^+ = \mathsf{IP}_{\mathsf{DB}}(\mathbf{z}^+, \mathcal{C}^+) \oplus \mathsf{IP}_{\mathsf{DB}}(\mathbf{z}^-, \mathcal{C}^-)$ and $\mathbf{ct}^- = \mathsf{IP}_{\mathsf{DB}}(\mathbf{z}^+, \mathcal{C}^-) \oplus \mathsf{IP}_{\mathsf{DB}}(\mathbf{z}^-, \mathcal{C}^+)$ by considering their sign. The subtraction $\mathbf{ct}^+ - \mathbf{ct}^-$ after decryption gives the desired inner product value with the encoding, and the final result can be obtained by running Decode.

Due to space constraints, we provide the full description of IDFace and its subroutine algorithms in Appendix F.

5. Analyses on IDFace

In this section, we provide several analyses of the proposed IDFace, along with the implementation result. We used an i7-11700K CPU, which has 3.60GHz processor speed with 8 cores and 16 threads. For AHEs in IDFace, we use two schemes: Paillier cryptosystem (PC) and CKKS. Since CKKS also satisfies the multiplicative homomorphic property, previous HE-based methods can be directly compared. For parameters of each AHE, we select a 2048-bit plaintext modulus for PC, and a 61-bit plaintext modulus, a 220-bit ciphertext modulus with 4096 message slots for CKKS. Each setting satisfies 112-bit and 128-bit security levels, respectively. We experimentally verified that CKKS preserves almost 50-bit precision in this setting, and thus we used only 50-bit for our Encode when using CKKS. We implemented our IDFace by exploiting IPCL [28] library for PC and HEaaN [13] library for CKKS. Each library is written in modern standard C++ and provides a Python wrapper.

5.1. Efficiency Analysis

We first provide a concrete efficiency analysis on IDFace.$\mathsf{IP}_{\mathsf{DB}}$ and introduce a further acceleration technique via the trade-off between efficiency and accuracy. Recall that our transformation T_β in IDFace.$\mathsf{IP}_{\mathsf{DB}}$ converts the given biometric template into a signed binary vector in $\mathcal{Z}_\beta^d$. Since the number of required additions in $\mathsf{IP}_{\mathsf{DB}}$ for binary vectors is the same as the number of nonzero components, $2(\beta - 1)$ ciphertext addition suffices during four iterations of execution of $\mathsf{IP}_{\mathsf{DB}}$. The Tab. 1 provides the comparison on the cost for identification on the database of size mN between a naïve method, some previous works [2, 3, 7, 20] tried to reduce the cost for inner product, and our IDFace.

Method	Add	Mult	Rotate	DEC	# Ct
Naïve	$mN(d-1)$	mN	$mN(d-1)$	mN	mN
[7]	$mN\lceil\log_2(d)\rceil$	mN	$mN\lceil\log_2(d)\rceil$	mN	mN
HERS [20]	$m(d-1)$	md	0	m	m
MFBR-v2 [3]	$mN\lceil\log_2(N)\rceil$	0	$mN\lceil\log_2(N)\rceil$	mN	mN
MFBR-ID [4]	$m(d-1)$	0	0	m	$8m$
IDFace	$2(\beta-1)$	0	0	2	2

Table 1. Comparison of the number of operations for identification and storage. m: the number of transformed templates in each message slot of AHE. mN: the number of enrolled templates, which are d-dimensional vectors. #Ct: the number of ciphertexts.

Note that the number m of templates handled by Encode is determined by the maximum value of inner product, which is $\min\{\alpha, \beta\}$. Hence, choosing smaller β while fixing α further accelerates IDFace.$\mathsf{IP}_{\mathsf{DB}}$. Of course, choosing smaller β leads to inaccurate inner product values. Since the effect of β on m depends on the length of β as a binary representation, we select $\beta = 63$ (resp. $\beta = 127$ or $\beta = 341$) while fixing α to 341. In this case, we can encode many templates at once up to 342 (resp. 293 or 243) in PC and 8 (resp. 7 or 5) in CKKS.

Implementation Results. We now verify the efficiency of our IDFace in our experimental environment. We note that every experiment in this subsection was done by generating random vectors. We first evaluate the time spent on the enrollment and identification for various numbers of enrolled identities. The number of identities varies from 2^{13} to 2^{20} ≈1 M, and we select $\beta = 63, 127, 341$ while α is fixed to 341. The results for each algorithm are illustrated in Fig. 2. Unlike PC, IDFace from CKKS can processes more identities at the same time because of the coefficient packing. Concretely, IDFace in our setting can process 20480, 28672, 32768 identities at once for $\beta = 341, 127, 63$, respectively. This is why the elapsed time for the number of identity 2^{13}–2^{15} in CKKS seems unchanged. In addition, the elapsed time in all parameter settings grows linearly with respect to the number of enrolled identities, because IDFace reduces the cost of inner product rather than that of searching. Nevertheless, in all parameter settings, the IDFace with CKKS can perform an identification in less than a second on the database of $2^{20} \approx 1$ M identities.

To compare our IDFace with other existing BTPs, we provide the computational cost of each method for enrollment and identification on a database of size 1 M. We re-

[2] We can use the different parameter β from α that is used for enrollment in order to take advantage of the trade-off between accuracy and efficiency. Detailed explanation and analysis will be provided in Section 5.

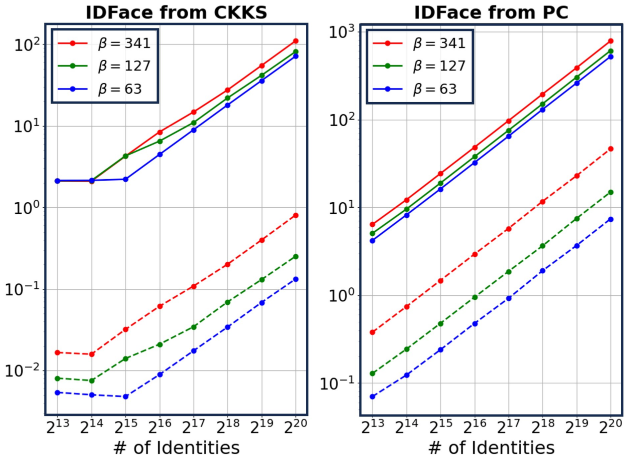

Figure 2. Elapsed time for Enroll and Identify on various numbers of enrolled identities. Solid line: elapsed time for Enroll. Dashed line: elapsed time for Identify.

implemented IronMask [40], SecureVector [51], [6, 7, 17], BlindMatch [11], HERS [20], MFBR-v2 [3] and MFBR-ID [4] in our experimental environment. We followed the pseudocodes provided in each original paper. Since [6, 7] and HERS exploited a dimensionality reduction technique for faster identification, we tested both cases when the dimension of the feature vector was 512 and 128[3]. For BlindMatch, we select the number of partitions of the templates (N_{in} in their paper) to be 8 and 16 for 128 and 512-dimension, respectively. Since MFBR-v2 and MFBR-ID are designed for BFV [21], we implemented them by OpenFHE v1.2.4 [1] in C++17 with the same FHE parameters according to their papers. Moreover, we measured the time spent on identification without BTP, *i.e.*, one matrix-vector multiplication without encryption.

Remark that some methods were omitted because their main efficiency gain came from other previous works [25, 34], or severe accuracy degradation is expected for large-scale databases of size 1 M [5, 18, 57]. In addition, we reported estimated results for some works [7, 17, 40] by first evaluating them on the database of 1,000 identities and multiplying 1,000 on the result. This estimation would be reasonable because the computational cost of theirs is linear to the number of identities. Finally, unlike other HE-based methods, IronMask does not require a secret key. Although comparing it with other methods, including ours, would be unfair, we added it as a reference for the efficiency analysis.

We also note that some prior works, including [6, 7, 11, 18] and HERS, support the case where the queried vectors are encrypted. Such a setting corresponds to the case where the privacy of the query vector against server(s) is crucial, e.g., outsourcing the data and matching process to 3rd party cloud servers [17, 20, 25]. Nevertheless, when secure storage of templates is the primary goal, such as in our target scenarios, considering an encrypted query vector would be an overkill. Therefore, for the sake of fair comparison, we implemented those schemes as if the query vector were not encrypted.

[3] Although their original papers provided results for lower dimensions, such as 64 or 32, we omitted them because of a notable accuracy degradation. The amount of degradation will be further discussed in Appendix D.5.

Method (Param.)	Enroll	Identify	Storage	Primitive
Non-protected	N/A	0.063 ± 0.0005s	2GB	N/A
IronMask [40]†	$4,416 \pm 3.73$s	97.9 ± 0.27s	1024GB	FC
SecureVector [51]	350 ± 4.09s	398 ± 2.75s	5GB	PC
[17]†,‡	917 ± 3.57s	$290,605 \pm 458$s	132GB	CKKS
[7] (512)†,‡	909 ± 3.30s	$6,210 \pm 16.5$s	132GB	CKKS
[7] (128)†,‡	846 ± 1.58s	$5,037 \pm 8.49$s	33GB	CKKS
[6] (512)‡	198 ± 0.73s	826 ± 1.14s	16.5GB	CKKS
[6] (128)‡	50 ± 0.05s	171 ± 0.25s	4.125GB	CKKS
BlindMatch (512)‡	253 ± 13.9s	233 ± 10.4s	16.5GB	CKKS
BlindMatch (128)‡	68.5 ± 2.60s	66.6 ± 1.79s	4.125GB	CKKS
HERS (512)‡	199 ± 0.68s	48.9 ± 0.20s	16.5GB	CKKS
HERS (128)‡	$\mathbf{49 \pm 0.05}$s	12.3 ± 0.04s	4.125GB	CKKS
MFBR-v2† [3]	1348 ± 2.32s	1026 ± 8.04s	131.1GB	BFV
MFBR-ID [4]	1641 ± 5.98s	0.545 ± 0.006s	132.1GB	BFV
IDFace (341)	751 ± 2.44s	43.4 ± 0.38s	2.25GB	PC
IDFace (127)	582 ± 0.37s	14.2 ± 0.13s	1.75GB	PC
IDFace (63)	501 ± 1.73s	7.08 ± 0.03s	**1.5GB**	PC
IDFace (341)	109 ± 0.10s	0.753 ± 0.005s	6.6GB	CKKS
IDFace (127)	80 ± 0.13s	0.232 ± 0.002s	4.71GB	CKKS
IDFace (63)	72 ± 0.11s	$\mathbf{0.126 \pm 0.001}$s	4.125GB	CKKS

Table 2. Time and storage cost comparison of identification protocols with 1 M enrolled identities. FC: Fuzzy Commitment. PC: Paillier Cryptosystem. †: Estimated from 1,000 identities. ‡: This scheme supports encrypted queries.

The results are given in Tab. 2. We reported the average elapsed time with a 1-σ error for 10 runs. Our IDFace significantly outperforms previous HE-based BTPs in terms of Notably, compared to HERS, with the parameter $\beta = 341$ (resp. $\beta = 127$ or $\beta = 63$) achieves $16.3\times$ (resp. $53.0\times$ or $97.6\times$) faster identification speed with only 60% (resp. 20% or 0%) extra storage. In addition, compared to MFBR-ID [4], which is another multiplication-free method based on look-up tables, IDFace with $\beta = 63$ spent $23.5\times$ and $4.5\times$ less time for enrollment and identification, respectively, while requiring $31.8\times$ smaller storage for protected templates. Furthermore, when comparing ours to a naïve identification without BTP, i.e., performing a matrix-vector multiplication between the enrolled and queried template(s), the overhead from both computational and storage costs of IDFace with $\beta = 63$ is only about $\mathbf{2}\times$. These results suggest that our IDFace could be deployed in application scenarios where real-time identification is required, such as at an airport or the entrance of a building.

5.2. Accuracy Analysis

We provide the benchmark result of IDFace on various datasets, including IJB-C [50], LFW [24], CFP-FP [59], and AgeDB [55]. For the feature extractor, we used the pre-trained AdaFace [38] with the IResNet101 backbone trained on the WebFace12M dataset [72], which is available at the CVLFace library [37]. Throughout this section, we refer to a non-protected feature extractor as a plain model.

Dataset	Metric (FAR)	Plain	IDFace, Parameters: (α, β)					
			$(512, 512)$	$(341, 341)$	$(127, 127)$	$(63, 63)$	$(341, 127)$	$(341, 63)$
LFW	Accuracy	99.82%	99.77%	99.78%	**99.83%**	99.73%	99.80%	99.82%
CFP-FP		**99.24%**	98.96%	99.24%	99.09%	98.20%	99.19%	98.99%
AgeDB		**98.00%**	97.67%	97.97%	97.55%	96.48%	97.80%	97.23%
IJB-C (V)	TAR (1e-3)	**98.39%**	97.93%	98.27%	97.90%	96.94%	98.14%	97.78%
IJB-C (ID)	TPIR (1e-2)	**96.42%**	95.15%	95.95%	95.32%	92.87%	95.72%	95.19%
	TPIR (1e-3)	85.81%	85.83%	85.11%	85.40%	86.21%	**86.59%**	85.96%
	TPIR (1e-4)	65.60%	63.88%	**68.38%**	62.38%	61.39%	63.45%	64.66%

Table 3. Benchmark results of non-protected feature extractor (Plain) and IDFace for various (α, β).

We evaluated the effect of our transformation parameters (α, β) on the accuracy of the FRS. In the first 4 columns of Tab. 3, we provide benchmark results when $\alpha = \beta$ with $\alpha = 63, 127, 341$ and 512. One notable result is that the result from $\alpha = 341$ is better than that of $\alpha = 512$ in most cases. Recall that we considered the setting with $\beta < \alpha$ for speed-up while allowing extra accuracy degradation. So, we evaluated the effect of the choice $\beta = 63, 127$ on accuracy while α is fixed to 341. The results for them are provided in the rightmost two columns of Tab. 3. We can figure out that IDFace shows less than 1% performance loss for LFW, CFP-FP, and AgeDB even for the setting $\beta = 63$. For our main interest, identification on IJB-C, a trade-off between query speed and performance was evident. But, for $\beta = 341$, the loss for TAR was still less than 1%.

Due to space constraints, we defer several detailed analyses results to Appendix. For example, in Appendix D.3, we provide results from other recent feature extractors, including ArcFace [16], MagFace [52], SphereFace2 [67], ElasticFace [8], and AdaFace-KPRPE [39], showing that the proposed IDFace shows consistently small accuracy degradation as in Tab. 3. In Appendix D.6, we provide our analyses on how our transformation affects the accuracy, along with our interpretation of data marked with red in Tab. 3.

5.3. Security Analysis

We analyze the security of IDFace according to standard security requirements for BTP [29]: *irreversibility*, *revocablity*, and *unlinkability*. Informally, each of them is stated as follows: Irreversibility requires that extracting biometric information from a protected template be computationally infeasible. Revocability requires that the protected template can be renewed by the same biometric without security loss. To satisfy unlinkability, protected templates from biometrics of the same identity should be independent; useful information cannot be extracted by comparing them.

For the threat model, we consider an attacker who can take the public key pk and the encrypted database stored in the local server, without knowing the secret key sk stored in the key server. This threat model reflects several practical situations, such as when the key server is a secure tamper-proof hardware that stores a secret key, or when multiple local servers exist as shown in Appendix H. In both cases, the key server can become more secure with additional protections. The attacker's goal is to extract biometric information from the encrypted database. Because our database is encrypted by AHE, this goal can be regarded as breaking the IND-CPA security of AHE. Note that the AHE scheme exploited in IDFace satisfies IND-CPA security. That is, even when the stored ciphertexts are revealed, they are computationally indistinguishable from random strings from the view of the adversary, so IDFace satisfies the irreversibility. In addition, because the encryption algorithms of each AHE are randomized, two ciphertexts from the same biometric source are computationally indistinguishable. Unlinkability clearly holds for the same reason as revocability. To sum up, our IDFace satisfies all these security requirements.

6. Conclusion and Further Directions

Protecting and exploiting biometric templates simultaneously is a challenging problem because many cryptographic tools are not tailored for processing biometric templates in a real-valued vector format. In this study, we propose two techniques to tackle this challenging problem, and then show the effectiveness of our techniques by presenting IDFace, a plug-and-play HE-based face template protection method for efficient and secure large-scale identification. With IDFace, we successfully identify a queried template over encrypted template databases of 1M identities in *less than a second*, which is only **2**$\times$ slower compared to the setting without protection. We believe that IDFace enables reliable and secure applications of face identification systems in various circumstances.

Throughout this study, we leave several interesting questions. First, our almost-isometric transformation is of independent interest. Designing another transformation with more accurate distance relationship preservation or applying it to other applications would be interesting future directions[4,5]. In addition, we considered only passive attackers for setting the threat model of IDFace. Extending IDFace to secure against active attackers is open, and investigating this would be an interesting future work.

[4] For example, we designed a secure two-party computation-based variant of IDFace using our transformation in Appendix G.

[5] We also provided the result of applying IDFace on speaker verification and fingerprint verification tasks in Appendix I.

Acknowledgements

We would like to thank the anonymous ICCV 2025 reviewers for their valuable feedback.

References

[1] Ahmad Al Badawi, Jack Bates, Flavio Bergamaschi, David Bruce Cousins, Saroja Erabelli, Nicholas Genise, Shai Halevi, Hamish Hunt, Andrey Kim, Yongwoo Lee, et al. Openfhe: Open-source fully homomorphic encryption library. In *proceedings of the 10th workshop on encrypted computing & applied homomorphic cryptography*, pages 53–63, 2022. 7

[2] Amina Bassit, Florian Hahn, Raymond Veldhuis, and Andreas Peter. Multiplication-free biometric recognition for faster processing under encryption. In *2022 IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–9. IEEE, 2022. 3, 6

[3] Amina Bassit, Florian FW Hahn, Raymond NJ Veldhuis, and Andreas Peter. Improved multiplication-free biometric recognition under encryption. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 6(3):314–325, 2023. 6, 7

[4] Amina Bassit, Florian Hahn, Raymond Veldhuis, and Andreas Peter. Practical biometric search under encryption: Meeting the nist runtime requirement without loss of accuracy. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2025. 3, 6, 7

[5] Pia Bauspieß, Marcel Grimmer, Cecilie Fougner, Damien Le Vasseur, Thomas Thaulow Stöcklin, Christian Rathgeb, Jascha Kolberg, Anamaria Costache, and Christoph Busch. Hebi: Homomorphically encrypted biometric indexing. In *2023 IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–10. IEEE, 2023. 2, 3, 7

[6] Pia Bauspieß, Jonas Olafsson, Jascha Kolberg, Pawel Drozdowski, Christian Rathgeb, and Christoph Busch. Improved homomorphically encrypted biometric identification using coefficient packing. In *2022 International Workshop on Biometrics and Forensics (IWBF)*, pages 1–6, 10.1109/IWBF55382.2022.9794523, 2022. IEEE. 2, 3, 7

[7] Vishnu Naresh Boddeti. Secure face matching using fully homomorphic encryption. In *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–10. IEEE, 2018. 1, 2, 3, 6, 7

[8] Fadi Boutros, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Elasticface: Elastic margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 1578–1587, 2022. 1, 3, 8

[9] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (leveled) fully homomorphic encryption without bootstrapping. *ACM Transactions on Computation Theory (TOCT)*, 6 (3):1–36, 2014. 3

[10] Jung Hee Cheon, Andrey Kim, Miran Kim, and Yongsoo Song. Homomorphic encryption for arithmetic of approximate numbers. In *International conference on the theory and application of cryptology and information security*, pages 409–437. Springer, 2017. 3, 5

[11] Hyunmin Choi, Jiwon Kim, Chiyoung Song, Simon S Woo, and Hyoungshick Kim. Blind-match: Efficient homomorphic encryption-based 1: N matching for privacy-preserving biometric identification. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 4423–4430, 2024. 1, 2, 3, 7

[12] Hyunmin Choi, Simon S Woo, and Hyoungshick Kim. Blind-touch: Homomorphic encryption-based distributed neural network inference for privacy-preserving fingerprint authentication. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 21976–21985, 2024. 1

[13] CRYPTOLAB. Heaan. `https://github.com/snucrypto/HEAAN`, 2023. 6

[14] Thao M Dang, Lam Tran, Thuc D Nguyen, and Deokjai Choi. Fehash: Full entropy hash for face template protection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 810–811, 2020. 2

[15] J Daugman. How iris recognition works. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):21–30, 2004. 1

[16] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019. 1, 3, 8

[17] P. Drozdowski, N. Buchmann, C. Rathgeb, M. Margraf, and C. Busch. On the application of homomorphic encryption to face identification. In *2019 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–5, 2019. 2, 3, 7

[18] Pawel Drozdowski, Fabian Stockhardt, Christian Rathgeb, Daile Osorio-Roig, and Christoph Busch. Feature fusion methods for indexing and retrieval of biometric data: Application to face recognition with privacy protection. *IEEE Access*, 9:139361–139378, 2021. 2, 3, 7

[19] Chi Nhan Duong, Thanh-Dat Truong, Khoa Luu, Kha Gia Quach, Hung Bui, and Kaushik Roy. Vec2face: Unveil human faces from their blackbox features in face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6132–6141, 2020. 1

[20] Joshua J Engelsma, Anil K Jain, and Vishnu Naresh Boddeti. Hers: Homomorphically encrypted representation search. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2022. 1, 2, 3, 4, 6, 7

[21] Junfeng Fan and Frederik Vercauteren. Somewhat practical fully homomorphic encryption. *Cryptology ePrint Archive*, 2012. 3, 7

[22] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 169–178, 2009. 1

[23] Feng Hao, Ross Anderson, and John Daugman. Combining crypto with biometrics effectively. *IEEE transactions on computers*, 55(9):1081–1088, 2006. 1

[24] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database

forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008. 2, 7

[25] Hai Huang and Luyao Wang. Efficient privacy-preserving face identification protocol. *IEEE Transactions on Services Computing*, pages 1–10, 2023. 2, 7

[26] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Quantized neural networks: Training neural networks with low precision weights and activations. *Journal of Machine Learning Research*, 18(187): 1–30, 2018. 5

[27] Alberto Ibarrondo, Hervé Chabanne, Vincent Despiegel, and Melek Önen. Grote: Group testing for privacy-preserving face identification. In *Proceedings of the Thirteenth ACM Conference on Data and Application Security and Privacy*, pages 117–128, 2023. 3

[28] INTEL. Icpl. `https://github.com/intel/pailliercryptolib_python`, 2023. 6

[29] ISO 24745:22. Information security, cybersecurity and privacy protection — biometric information protection. Standard, International Organization for Standardization, Geneva, CH, 2022. 1, 8

[30] Xi Jia, Jiancan Zhou, Linlin Shen, Jinming Duan, et al. Unitsface: Unified threshold integrated sample-to-sample loss for face recognition. *Advances in Neural Information Processing Systems*, 36:32732–32747, 2023. 1, 3

[31] Shuaifan Jin, He Wang, Zhibo Wang, Feng Xiao, Jiahui Hu, Yuan He, Wenwen Zhang, Zhongjie Ba, Weijie Fang, Shuhong Yuan, et al. {FaceObfuscator}: Defending deep learning-based privacy attacks with gradient descent-resistant features in face recognition. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 6849–6866, 2024. 2

[32] Zhe Jin, Jung Yeon Hwang, Yen-Lung Lai, Soohyung Kim, and Andrew Beng Jin Teoh. Ranking-based locality sensitive hashing-enabled cancelable biometrics: Index-of-max hashing. *IEEE Transactions on Information Forensics and Security*, 13(2):393–407, 2017. 1

[33] Arun Kumar Jindal, Srinivasa Rao Chalamala, and Santosh Kumar Jami. Securing face templates using deep convolutional neural network and random projection. In *2019 IEEE International Conference on Consumer Electronics (ICCE)*, pages 1–6. IEEE, 2019. 2

[34] Arun Kumar Jindal, Imtiyazuddin Shaik, Vasudha Vasudha, Srinivasa Rao Chalamala, Rajan Ma, and Sachin Lodha. Secure and privacy preserving method for biometric template protection using fully homomorphic encryption. In *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 1127–1134, 2020. 2, 3, 7

[35] Yoon Gyo Jung, Jaewoo Park, Xingbo Dong, Hojin Park, Andrew Beng Jin Teoh, and Octavia Camps. Face reconstruction transfer attack as out-of-distribution generalization. In *European Conference on Computer Vision*, pages 396–413. Springer, 2024. 1

[36] Manuel Kansy, Anton Raël, Graziana Mignone, Jacek Naruniec, Christopher Schroers, Markus Gross, and Romann M Weber. Controllable inversion of black-box face recognition models via diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3167–3177, 2023. 1

[37] Minchul Kim. Cvlface: High-performance face recognition all-in-one toolkit. accessed: 2024-07-01. 7

[38] Minchul Kim, Anil K Jain, and Xiaoming Liu. Adaface: Quality adaptive margin for face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18750–18759, 2022. 1, 3, 7

[39] Minchul Kim, Yiyang Su, Feng Liu, Anil Jain, and Xiaoming Liu. Keypoint relative position encoding for face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 244–255, 2024. 1, 3, 8

[40] Sunpill Kim, Yunseong Jeong, Jinsu Kim, Jungkon Kim, Hyung Tae Lee, and Jae Hong Seo. Ironmask: Modular architecture for protecting deep face template. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16125–16134, 2021. 1, 2, 5, 7

[41] Sunpill Kim, Yong Kiam Tan, Bora Jeong, Soumik Mondal, Khin Mi Mi Aung, and Jae Hong Seo. Scores tell everything about bob: Non-adaptive face reconstruction on face recognition systems. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 1684–1702. IEEE, 2024. 1

[42] Arun Kumar Jindal, Srinivas Chalamala, and Santosh Kumar Jami. Face template protection using deep convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 462–470, 2018. 2

[43] Rohit Kumar Pandey, Yingbo Zhou, Bhargava Urala Kota, and Venu Govindaraju. Deep secure encoding for face template protection. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 9–15, 2016. 2

[44] Yenlung Lai, Zhe Jin, KokSheik Wong, and Massimo Tistarelli. Efficient known-sample attack for distance-preserving hashing biometric template protection schemes. *IEEE Transactions on Information Forensics and Security*, 16:3170–3185, 2021. 1

[45] Yen-Lung Lai, Zhe Jin, Andrew Beng Jin Teoh, Bok-Min Goi, Wun-She Yap, Tong-Yuen Chai, and Christian Rathgeb. Cancellable iris template generation based on indexing-first-one hashing. *Pattern Recognition*, 64:105–117, 2017. 1

[46] Yue Li, Wenrui Ding, Chunlei Liu, Baochang Zhang, and Guodong Guo. Trq: Ternary neural networks with residual quantization. In *Proceedings of the AAAI conference on artificial intelligence*, pages 8538–8546, 2021. 5

[47] Shuming Ma, Hongyu Wang, Lingxiao Ma, Lei Wang, Wenhui Wang, Shaohan Huang, Li Dong, Ruiping Wang, Jilong Xue, and Furu Wei. The era of 1-bit llms: All large language models are in 1.58 bits. *arXiv preprint arXiv:2402.17764*, 2024. 5

[48] Guangcan Mai, Kai Cao, Pong C Yuen, and Anil K Jain. On the reconstruction of face images from deep face templates. *IEEE transactions on pattern analysis and machine intelligence*, 41(5):1188–1202, 2018. 1

[49] Davide Maltoni, Dario Maio, Anil K Jain, Salil Prabhakar, et al. *Handbook of fingerprint recognition*. Springer, 2009. 1

[50] Brianna Maze, Jocelyn Adams, James A Duncan, Nathan Kalka, Tim Miller, Charles Otto, Anil K Jain, W Tyler Niggel, Janet Anderson, Jordan Cheney, et al. Iarpa janus benchmark-c: Face dataset and protocol. In *2018 international conference on biometrics (ICB)*, pages 158–165. IEEE, 2018. 2, 7

[51] Qiang Meng and Feng Zhou. Towards privacy-preserving, real-time and lossless feature matching. *arXiv preprint arXiv:2208.00214*, 2022. 7

[52] Qiang Meng, Shichao Zhao, Zhida Huang, and Feng Zhou. Magface: A universal representation for face recognition and quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14225–14234, 2021. 1, 3, 8

[53] Yuxi Mi, Zhizhou Zhong, Yuge Huang, Jiazhen Ji, Jianqing Xu, Jun Wang, Shaoming Wang, Shouhong Ding, and Shuigeng Zhou. Privacy-preserving face recognition using trainable feature subtraction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 297–307, 2024. 2

[54] Deen Dayal Mohan, Nishant Sankaran, Sergey Tulyakov, Srirangaraj Setlur, and Venu Govindaraju. Significant feature based representation for template protection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 2389–2396, 2019. 2

[55] Stylianos Moschoglou, Athanasios Papaioannou, Christos Sagonas, Jiankang Deng, Irene Kotsia, and Stefanos Zafeiriou. Agedb: the first manually collected, in-the-wild age database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 51–59, 2017. 2, 7

[56] Karthik Nandakumar, Anil K Jain, and Sharath Pankanti. Fingerprint-based fuzzy vault: Implementation and performance. *IEEE transactions on information forensics and security*, 2(4):744–757, 2007. 1

[57] Dailé Osorio-Roig, Christian Rathgeb, Pawel Drozdowski, and Christoph Busch. Stable hash generation for efficient privacy-preserving face identification. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 4(3):333–348, 2022. 2, 3, 7

[58] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *International conference on the theory and applications of cryptographic techniques*, pages 223–238. Springer, 1999. 3, 5

[59] Soumyadip Sengupta, Jun-Cheng Chen, Carlos Castillo, Vishal M Patel, Rama Chellappa, and David W Jacobs. Frontal to profile face verification in the wild. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9. IEEE, 2016. 2, 7

[60] Hatef Otroshi Shahreza and Sébastien Marcel. Face reconstruction from facial templates by learning latent space of a generator network. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 1

[61] Hatef Otroshi Shahreza and Sébastien Marcel. Template inversion attack against face recognition systems using 3d face reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19662–19672, 2023.

[62] Hatef Otroshi Shahreza, Anjith George, and Sébastien Marcel. Face reconstruction from face embeddings using adapter to a face foundation model. *arXiv preprint arXiv:2411.03960*, 2024. 1

[63] Veeru Talreja, Matthew C Valenti, and Nasser M Nasrabadi. Zero-shot deep hashing and neural network based error correction for face template protection. In *2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–10. IEEE, 2019. 2

[64] Pim Tuyls, Anton HM Akkermans, Tom AM Kevenaar, Geert-Jan Schrijen, Asker M Bazen, and Raimond NJ Veldhuis. Practical biometric authentication with template protection. In *Audio-and Video-Based Biometric Person Authentication: 5th International Conference, AVBPA 2005, Hilton Rye Town, NY, USA, July 20-22, 2005. Proceedings 5*, pages 436–446. Springer, 2005. 1

[65] Paul Voigt and Axel Von dem Bussche. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 10(3152676):10–5555, 2017. 1

[66] Zhibo Wang, He Wang, Shuaifan Jin, Wenwen Zhang, Jiahui Hu, Yan Wang, Peng Sun, Wei Yuan, Kaixin Liu, and Kui Ren. Privacy-preserving adversarial facial features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8212–8221, 2023. 2

[67] Yandong Wen, Weiyang Liu, Adrian Weller, Bhiksha Raj, and Rita Singh. Sphereface2: Binary classification is all you need for deep face recognition. In *International Conference on Learning Representations*, 2022. 1, 3, 8

[68] Emily Wenger, Francesca Falzon, Josephine Passananti, Haitao Zheng, and Ben Y Zhao. Assessing privacy risks from feature vector reconstruction attacks. *arXiv preprint arXiv:2202.05760*, 2022. 1

[69] Shenglin Yang and Ingrid Verbauwhede. Secure iris verification. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, pages II–133. IEEE, 2007. 1

[70] Jiancan Zhou, Xi Jia, Qiufu Li, Linlin Shen, and Jinming Duan. Uniface: Unified cross-entropy loss for deep face recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20730–20739, 2023. 1, 3

[71] Chenzhuo Zhu, Song Han, Huizi Mao, and William J Dally. Trained ternary quantization. In *International Conference on Learning Representations*, 2017. 5

[72] Zheng Zhu, Guan Huang, Jiankang Deng, Yun Ye, Junjie Huang, Xinze Chen, Jiagang Zhu, Tian Yang, Jiwen Lu, Dalong Du, et al. Webface260m: A benchmark unveiling the power of million-scale deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10492–10502, 2021. 7