Towards Secure Biometric Solutions: Enhancing Facial Recognition while Protecting User Data

Jose Silva[1,2][a], Aniana Cruz[1][b], Bruno Sousa[2][c], and Nuno Gonçalves[1][d]
[1]*Institute of Systems and Robotics - University of Coimbra (ISR-UC), Coimbra, Portugal*
[2]*Centre for Informatics and Systems of the University of Coimbra (CISUC), Coimbra, Portugal*
jose.silva@isr.uc.pt, anianabrito@isr.uc.pt, bmsousa@dei.uc.pt, nunogon@deec.uc.pt

Keywords: Facial recognition, Multi-factor authentication, Data privacy, Biometric templates, Cryptographic algorithms

Abstract: This paper presents a novel approach to the storage of facial images in databases designed for biometric authentication, with a primary focus on user privacy. Biometric template protection encompasses a variety of techniques aimed at safeguarding users' biometric information. Generally, these methods involve the application of transformations and distortions to sensitive data. However, such alterations can frequently result in diminished accuracy within recognition systems. We propose a deformation process to generate temporary codes that facilitate the verification of registered biometric features. Subsequently, facial recognition is performed on these registered features in conjunction with new samples. The primary advantage of this approach is the elimination of the need to store facial images within application databases, thereby enhancing user privacy while maintaining high recognition accuracy. Evaluations conducted using several benchmark datasets - including AgeDB-30, CALFW, CPLFW, LFW, RFW, XQLFW - demonstrate that our proposed approach preserves the accuracy of the biometric system. Furthermore, it mitigates the necessity for applications to retain any biometric data, images, or sensitive information that could jeopardize users' identities in the event of a data breach. The solution code, benchmark execution, and demo are available at: https://bc1607.github.io/FRS-ProtectingData.

1 INTRODUCTION

The era of digitization has brought several opportunities, as well as challenges and concerns, especially when it comes to the security and privacy of personal data. As the use of Multi-Factor Authentication (MFA) grows, the integration of technologies like facial recognition is becoming more prevalent. This trend gives rise to crucial inquiries regarding the safeguarding, utilization, and implementation of security and privacy measures for biometric data. In the initial quarter of 2024, critical vulnerabilities were identified in prominent software systems such as Fortinet's FortiOS (CVE-2024-21762) [Bahmanisangesari, 2024], Jenkins (CVE-2024-23897) [Gioacchini et al., 2024], and XZ Utils (CVE-2024-3094) [Wu et al.,]. These vulnerabilities were officially listed in the Common Vulnerabilities and Exposures (CVE) database with a Common Vulnerability Scoring System (CVSS) score of 9.8 or higher, meaning the severity of these security risks. Despite being different software programs, these cases pose threats to information privacy as they can be exploited to gain access to internal data or to execute unauthorized commands.

Facial recognition systems (FRS) exhibit vulnerabilities akin to those found in various software applications, as their design and implementation may encompass inherent flaws that are susceptible to exploitation [Abdullahi et al., 2024]. Notably, these systems are particularly vulnerable to presentation attacks and spoofing techniques, which exploit their intrinsic limitations and may lead to erroneous decision-making processes [Marcel et al., 2023]. Furthermore, the integrity of facial recognition systems, as well as other systems that depend on them, can be compromised by malicious alterations to the underlying databases. This underscores that vulnerabilities may arise not only from the algorithms and methodologies employed but also from the data management practices in place [Mousa et al., 2020]. Consequently, decisions regarding the selection of biometric features

[a] https://orcid.org/0000-0002-7114-7777
[b] https://orcid.org/0000-0001-5420-6651
[c] https://orcid.org/0000-0002-5907-5790
[d] https://orcid.org/0000-0002-1854-049X

for database storage, the required accuracy thresholds, and the established procedural frameworks are crucial for mitigating recognition errors and addressing the growing concerns related to data privacy [Lala et al., 2021]. This paper proposes a comprehensive approach that emphasizes considerations of database privacy alongside the necessity for an appropriate level of accuracy.

The aim of this paper is to propose an innovative approach to facial recognition technology for authentication purposes in everyday applications, with a primary focus on maximizing accuracy while implementing a MFA framework. One of the foundational principles guiding this design is the elimination of the need to store biometric data in raw or plaintext formats, which could enable unauthorized recognition of individuals beyond the confines of the application domain. To enhance security in the authentication process, it is imperative that biometric authentication operates at high levels of accuracy; consequently, the methodology presented in this paper is designed to maintain, if not improve, the accuracy of recognition and authentication processes. Furthermore, the proposed approach must be inherently scalable, facilitating the integration of various standardized data protection regulations pertaining to encryption.

Biometric characteristics refer to unique physical or behavioral traits of an individual, such as facial features, retina patterns, signatures, or typing patterns [Delac and Grgic, 2004]. Biometric Template (BT) are digital representations created by extracting and encoding biometric features [Sarkar and Singh, 2020]. In the scientific literature, Biometric Template Protection (BTP) algorithms can be categorized into image-level and feature-level approaches.

In image-level BTP, techniques such as distortion functions [Kirchgasser et al., 2020], morphing cancellation (morphed cancellable face images) [Ratha et al., 2001], block scrambling [Dong et al., 2019], XOR operations, permutations, and filters are used to alter the original image (visual secret sharing) [Manisha and Kumar, 2020]. These methods can also incorporate revocability through user-defined passwords in MFA systems [Singh et al., 2021]. Additionally, images can be protected by employing co-occurrence matrices to obtain distinct features [Dabbah et al., 2007]. In this case, it is not possible to correlate the final image with the initial one [Dang and Choi, 2020].

On the other hand, the majority of BTP methods operate in the feature-level domain, utilizing cancelable functions or cryptosystems. For example, some methods rely on cryptographic functions like Homomorphic Encryption (HE) or employ hash functions. Approaches in [Ma et al., 2017, Boddeti, 2018, Drozdowski et al., 2019, Jindal et al., 2020, Drozdowski et al., 2021a, Drozdowski et al., 2021b, Engelsma et al., 2022, Osorio-Roig et al., 2021] encrypt the features using HE, enabling comparison in the encrypted domain. However, similar to other encryption algorithms, features can be decrypted with knowledge of the encryption key [Hahn and Marcel, 2022a]. Algorithms based on hash functions are applied to stable objects, derived mathematically from features (e.g. fuzzy commitment scheme [Juels and Wattenberg, 1999]).

In this paper, we propose a method that adheres to the properties of BTP as delineated in the literature: irreversibility, revocability, and unlinkability [Ramu and Arivoli, 2012]. To leverage artificial intelligence and facial image discriminators, we adopt a feature-level approach. For privacy considerations, our methodology employs standard and secure cryptographic techniques, including hash functions (SHA-512), encryption algorithms (Advanced Encryption Standard (AES) [National Inst Of Standards And Technology Gaithersburg Md, 2001]), Pseudorandom Number Generator (PRNG), and Time-based one-time password (TOTP). Our approach incorporates MFA that combines TOTP authenticators with facial recognition algorithms, specifically Convolutional Neural Network (CNN), along with distance and similarity functions. Moreover, we implement a process to protect biometric data through hashing and encryption techniques. This method enables the generation of secure objects that can subsequently be stored in application databases. To enhance accuracy levels, our solution makes facial recognition decisions based on the domain of biometric features extracted by the CNN.

In our approach, the irreversibility, revocability, and unlinkability of biometric characteristics are achieved by creating computationally secure cryptographic objects designed to be difficult to reverse, revocable, and capable of generating multiple instances from the same facial image, while ensuring unlinkability so that they do not inherently identify any individual. An implementation of this proposal was developed using Python libraries and evaluated against established benchmarks available in the scientific literature for research purposes, namely the Labeled Faces in the Wild (LFW) [Huang et al., 2008], Age Database 30 (AgeDB-30) [Moschoglou et al., 2017], Cross-Age LFW (CALFW) [Zheng et al., 2017], Crosspose LFW (CPLFW) [Zheng and Deng, 2018], Racial Faces in the Wild (RFW) [Wang et al., 2019], and CrossQuality Labeled Faces in the Wild (XQLFW) [Knoche et al., 2021].

The benchmarks presented herein comprise a series of tests along with their respective outcomes. The CNN model employed in this study is known as the MagFace model, as proposed by Meng et al. [Meng et al., 2021]. The MagFace model was trained utilizing the MSIM-V2 dataset. The facial recognition model functions as a black box, utilized solely for the extraction of feature vectors (embeddings). The primary Python libraries employed in the implementation include *binascii*, *random*, *cryptography*, *secrets*, *hashlib*, and *pyotp*.

The primary contributions of this paper focus on the design of an approach and methodology aimed at fulfilling four essential quality requirements. Specifically, our approach has been crafted to satisfy the following criteria: (1) MFA integrated with facial recognition; (2) user registration through cryptographic objects that ensure irreversibility, revocability, and unlinkability; (3) the creation of a database that does not contain biometric data while still enabling the confirmation and assurance of accurate facial authentication; and (4) the solution must not compromise the accuracy of the FRS defined by the CNN. In this paper, we implement the proposed approach and conduct benchmarks using available tests and metrics to evaluate a FRS, as outlined in the scientific literature. This evaluation aims to analyze and ascertain whether our proposed solution meets the established objectives.

The Section 2 addresses the relevant literature and prior research essential for the formulation of the proposed approach.

2 RELATED WORK

FRS must adhere to various quality attributes, including accuracy, computational efficiency, security, privacy, and usability [Hahn and Marcel, 2022b, Ekka et al., 2022]. In our approach, the primary focus lies on privacy, with data confidentiality being of utmost importance. Thus, in this study, methods that incorporate BTP at the feature level using cryptographic systems or utilizing cancelable functions are considered appropriate. Generally, BTP aims to achieve properties such as accuracy, irreversibility, renewability, and unlinkability [Hahn and Marcel, 2022b]. Rui et al. has introduced the Mission Success Rate property as a new addition to the biometric privacy criteria [Rui and Yan, 2019]. This property focuses on the system's ability to withstand attacks while also maintaining the confidentiality of biometric data, in addition to the existing properties of irreversibility, renewability, and unlinkability.

In [Li and Kot, 2010], a fingerprint authentication system is proposed which utilizes data hiding and embedding techniques to securely conceal private user information within a fingerprint template. In the registration process, a user's identity is embedded within their unique fingerprint template. This template, containing the encrypted data, is subsequently stored in a database for authentication purposes. This procedure outlines a methodology for concealing user data within images using steganography. In [Li and Kot, 2012], the template is generated by combining two fingerprints in a way that extracting a single fingerprint would be computationally challenging. Random and dynamic identifiers can be utilized for the purpose of associating and disassociating users with random objects. These random objects help to introduce entropy when combined with biometric characteristics.

In [Dang and Choi, 2020, Smith and Xu, 2011], the face-based key generation approach is described. This approach is a deterministic procedure that ensures zero-uncertainty key generation by leveraging auxiliary data storage. The key generation process involves preprocessing, distorting, and extracting facial features from the photograph, followed by utilizing randomization to construct a stable template and ultimately generate the key. This process is similar to generating a hash, with the distinction that it allows for intra-class variances (variations within images of the same face). The distortion step involves the application of cancelable functions before feature extraction, while the randomization step involves introducing entropy to the generated data.

In [El-Shafai et al., 2021], a novel authentication framework based on a genetic encryption algorithm is proposed. This algorithm takes an image and generates a cancelable biometric image. The algorithm utilizes permutation matrices, random number generator functions, divides the image into parts, processes the parts, applies crossover and mutation operations repeatedly. The final output hides the discriminative features of the biometric templates. It also achieves a high accuracy, with an average Area Under the Curve (AUC) of 0.9998. For authentication purposes, the database will store the cancelable biometric images. This approach is not exactly what is desired for our proposal, as it involves storing cancelable biometric images in a database. Additionally, it is important in our approach to have direct access to the face photograph for applying active liveness detection (involves requiring the user to perform a specific action, ensuring their active participation in the authentication process) and validating the International Civil Aviation Organization (ICAO) security list. The International Organization for Standardization (ISO) 19794-5 standard[1] outlines rules for taking a passport-style face photograph.

There are several approaches in the literature that utilize BTP algorithms following the extraction of features. This preference can be attributed to the availability of various pre-trained neural networks that have demonstrated the ability to extract and discriminating facial features from face images [Hahn and Marcel, 2022a]. Some examples of neural networks that act as feature discriminators and have an implemented version available are the Inception ResNet model [Schroff et al., 2015], the ResNet 50 model (ArcFace) [Deng et al., 2019], the QualFace model [Tremoço et al., 2021], the Idiap model [Hahn and Marcel, 2022b], and MagFace [Meng et al., 2021]. The MagFace model was selected for its high levels of accuracy.

```
Data: 1999
Result: Recognition system decision
protected BT = registration BT − codeword;
codeword' = validation BT − protected BT;
if hash(codeword) == hash(codeword') then
    The recognition is successful;
else
    The recognition is not successful;
end
```

Algorithm 1: Fuzzy Commitment scheme

After feature extraction, various BTP algorithms can be applied, such as the Fuzzy Commitment scheme [Juels and Wattenberg, 1999]. The Fuzzy Commitment scheme is a recognition procedure involving registration and validation phases. During the registration phase, a codeword is generated and linked with the user. A codeword is a value that is used to achieve a certain security goal, such as encryption, decryption, or authentication. The difference between the registration BT and the codeword produces the protected BT. The BT registration, along with the hash of the codeword, is stored in the database. In the validation phase, the aim is to recover the codeword' executing the inverse operation. If the retrieved hash of the codeword' matches the stored hash, the recognition is successful (see alg. 1). Error correction functions can rectify a certain number of errors in the codeword', but the hashes must match precisely. This algorithm was found to be insecure, as subsequent studies revealed the possibility of reversing the process and obtaining the template without knowledge of the codeword [Keller et al., 2020,Keller et al., 2021, Hahn and Marcel, 2022a]. Nevertheless, the essence of this method is to present a challenge where the user must provide a validation object identical to the registration object.

3 RESEARCH PROPOSAL

This section outlines a facial authentication approach that employs MFA and ensures the irreversibility, revocability, and unlinkability of registration objects. It advocates for a database devoid of biometric data, facilitating secure and effective authentication via CNN-defined FRS. The approach enhances existing systems by adding validation steps to improve the security and privacy of biometric data both in transit and within stored databases.

In the following Subsection 3.1, we will present a detailed overview of the proposed system, which encompasses two primary functional requirements: the **Registration Phase** and the **Validation Phase**. These phases involve interactions between the **Client** and two designated servers responsible for biometric recognition and authentication: the **Registration Server** and the **Authentication Server**. This interaction is illustrated in Fig. 1.

3.1 Overview of the approach

Our facial authentication system consists of three key entities: the Client, the Registration server, and the Authentication server. The system operates in two distinct phases: the Registration Phase (RP) and the Validation Phase (VP). Below, we present these two phases.

The RP begins when a Client makes a request to the Registration server. The Client submits a facial image, referred to as Frame 1 (F_1). The Registration server validates F_1 against a security checklist, which may include requirements outlined by the International Civil Aviation Organization (ICAO). If F_1 is determined to represent a valid and coherent human face, the Registration of the client proceeds. This Registration process entails the creation of several cryptographic objects, including the *username*, *opt_key*, *biometric_key*, *user_hash*, *user_key*, and authentication *proof*, which will be elaborated upon in Sections 3.2 and 3.3.

The Registration and Authentication servers communicate, allowing the Authentication server to store the relevant cryptographic objects. Subsequently, the Registration server shares the cryptographic objects with the Client, excluding the *user_hash*, the *user_key*, and the *proof*, which remain solely with the servers. The RP concludes when the Registration Server deletes F_1 and all previously generated crypto-

[1] https://www.iso.org/standard/50867.html

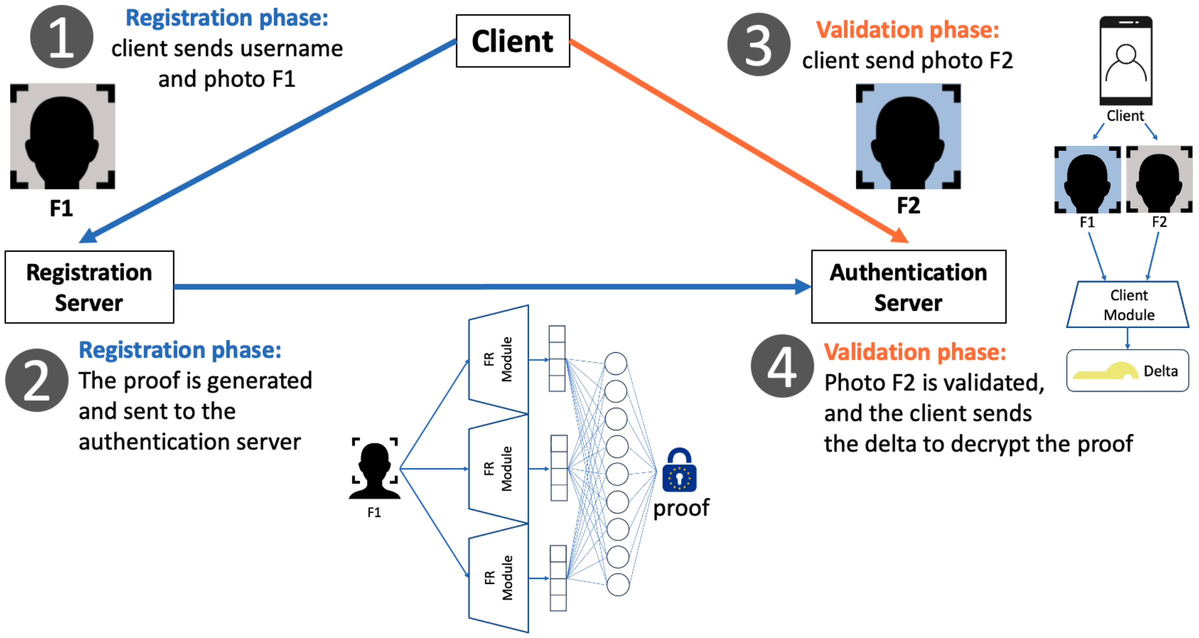

Figure 1: Architecture of the proposed approach: an illustration of the Registration (blue) and the Validation Phase (orange).

graphic objects, ensuring it retains no persistent memory of these items.

The Authentication server and the Client do not have the responsibility to generate cryptographic keys (e.g., *opt_key*, *biometric_key*, *user_key*). This responsibility is delegated to the Registration server, which selects appropriate and secure cryptographic functions for generating these objects. These objects, including *user_key* and authentication *proof*, are unique to each registration and exist solely on the server side. The authentication *proof* is crucial for authentication during the validation phase, which relies on hashing functions. This approach utilizes hashing to prevent the storage of biometric data on the server side, thereby mitigating the risk of exposing sensitive biometric information from the client's facial data F_1. In the event of a database breach, it should not be possible to retrieve any biometric information from the authentication *proof*. Consequently, the photograph used for registration, F_1, may be reused to create new and unique cryptographic objects.

Table 1 presents the data recorded by each entity upon the completion of the RP.

The VP occurs between the Client and the Authentication server. In a secure manner, the Client submits their *username* and a new facial image, referred to as Frame 2 (F_2). Initially, authentication is conducted through a challenge presented by the Authentication server, wherein the Client proves their identity using the *opt_key*, resulting in Proof 1 (P_1).

Table 1: Attributes required in the registration phase.

Parameters	Registration	Authentication	Client
F_1	×	×	✓
username	×	✓	✓
otp_key	×	✓	✓
biometric_key	×	✓	✓
user_hash	×	✓	×
user_key	×	✓	×
proof	×	✓	×

Subsequently, the Client generates Proof 2 (P_2) by extracting biometric features from F_2 and utilizing their *biometric_key*, which was obtained during the RP. The Authentication server utilizes P_2 to verify that the Client has accurately extracted the biometric features from F_2 and applied their *biometric_key*. Following this, the Client produces a parameter that we designated by *delta*, which enables the Authentication server to recalculate the registered biometric features (through mathematical functions detailed in Section 3.3). The Authentication server then processes these biometric features and decrypts the proof to validate its integrity. Upon successful validation, the Client will be authenticated if the biometric templates from Frame 1 (F_1) and Frame 2 (F_2) are found to be identical, exceeding a predefined threshold.

In the literature, various proposals for biometric recognition and authentication utilize biometric templates, applying distance or similarity functions. Our approach also validates similarity; however, it strategically incorporates cryptographic functions to ensure that biometric templates are neither stored nor transmitted over the network. Table 2 presents the parameters required during the VP.

For authentication to be considered valid, four conditions must be met: (1) the biometric registration features, BT_1, and the biometric validation features, BT_2 are similar, as the similarity function's result exceeds the threshold; (2) the Client presents P_2 to demonstrate that they have created a valid protected version of the biometric validation template, BTP_2, using their unique $biometric\_key$; (3) the Authentication server utilizes the $delta$ to obtain the biometric registration features BT_1. Subsequently, it recalculates the protected version of the biometric registration template BTP_1, since both BT_1 and BTP_1 are not stored in the server database; (4) the Authentication server utilizes BTP_1 to decrypt and validate the contents of the authentication $proof$.

Table 2: Attributes required in the validation phase.

Parameters	Authentication	Client
$username$	✓	✓
otp_key	✓	✓
P_1	✓	✓
$biometric\_key$	✓	✓
F_1	×	✓
BT_1	×	✓
BTP_1	×	✓
F_2	✓	✓
BT_2	×	✓
BTP_2	×	✓
$delta$	✓	✓
P_2	✓	✓
BT_2'	✓	×
BTP_2'	✓	×
BTP_1'	✓	×
$proof$	✓	×
$user\_hash$	✓	×
$user\_key$	✓	×
BT_1'	✓	×
$threshold$	✓	×

The Subsections 3.2 and 3.3 detail this approach, including the algorithm and the messages exchanged between entities during the registration and validation phases, respectively.

3.2 Registration Phase

Clients and Authentication servers rely on Registration servers for critical functions in recognition and authentication processes. The Registration servers evaluate whether the submitted images meet the necessary security requirements and generate reliable proofs for authentication. Clients trust Registration servers to protect their biometric data, while Authenticators depend on these servers to verify that a submitted image is a true representation of the client. As a result, both Clients and Authenticators have a vested interest in ensuring transparency in the operational procedures of Registration servers and supporting the adoption of open-source code.

In the RP, the Registration server will need a set of random parameters to decompose biometric data and transform it into a fixed code, referred to as the authentication $proof$. To achieve this, the Registration server utilizes a PRNG to generate the parameters known as $biometric\_key$, otp_key and $user\_key$ in a pseudo-random manner.

The RP begins when the Client submits an image, denoted as F_1. This image undergoes processing to recognize the face and extract biometric features. An algorithm is then applied to evaluate whether the image meets several required criteria (ICAO), including sufficient lighting, absence of shadows, full visibility of the face, and acceptable facial expressions. This assessment is conducted using neural networks that validate the client's face and ultimately generate a vector containing the extracted facial characteristics. Subsequently, from F_1, a feature vector BT_1 comprising 1024 floats is produced using a CNN. This embedding encapsulates the sensitive biometric information that we aim to protect in this study.

To safeguard biometric characteristics, we employ a decomposition and transformation process to obfuscate the biometric embeddings. The resulting values are decomposed, shuffled, and then combined with a $biometric\_key$, yielding a fixed-size code. This process functions similarly to a hash function; any modification to either the biometric template BT_1 or the $biometric\_key$ results in a distinctly different output. Nevertheless, the data remains transformed, allowing for subsequent facial recognition within this modified domain, which contrasts with traditional hash functions. In this context, we utilize the Biometric Template Protection Function known as PolyProtect, as proposed in [Hahn and Marcel, 2022b], to generate cryptographic keys and initialization vectors for the AES-256 symmetric encryption algorithm used in this study. This deterministic implementation ensures that identical inputs consistently produce the same output. This design choice is intentional, as it aims to maintain the accuracy of the recognition system, which will be evaluated in Section 4.

In our implementation, the function accepts a feature vector BT_1 and a $biometric\_key$ as inputs to generate a cryptographic key and an initialization vector (IV) for the encryption process. This protection function, similar to PolyProtect [Hahn and Marcel, 2022b], irreversibly distorts biometric information, thereby increasing its entropy and transforming it into a fixed code that deviates from the distribution of the original biometric features. We have implemented a modified version of the PolyProtect algorithm. Initially, the embedding V, which contains 1024 values, is divided into 32 non-overlapping sets ($m = 32$), analogous to the case in which there is zero overlap in the PolyProtect algorithm [Hahn and Marcel, 2022b]. The values of the embedding are then mapped using random coefficients c and exponents e, which are derived from the *biometric_key* generated by the PRNG.

A total of 32 terms are generated from a vector of 1024 values. Equation 1 corresponds to the generation of the first two terms, where V represents the initial embedding. For each subsequent term, the most significant digit of the float value is retained, while the remaining digits are discarded. For instance, if a float value is 0.00005678, the digit '5' is retained, and all other digits are disregarded. In cases where the selected digit is zero, a random digit is generated using the PRNG, with the *biometric_key* serving as the seed. These selected digits are then concatenated to form a key and IV for the AES-256 encryption algorithm.

$$T_1 = \sum_{i=1}^{m} c_i \cdot V_i^{e_i} \ , \ T_2 = \sum_{i=m+1}^{2m} c_i \cdot V_i^{e_i} \tag{1}$$

The Registration server extracts the biometric characteristics, generating a biometric template BT_1 (eq.2), which is then transformed and canceled to create the protected biometric template BTP_1 (eq.3). This transformation is conducted through our Biometric Template Protection method, BTP, initialized with the *biometric_key* (based on the PolyProtect strategy).

$$\text{CNN}(valid\_frame) = BT_1 \tag{2}$$

$$\text{BTP}(biometric\_key, BT_1) = BTP_1 \tag{3}$$

A hash function is applied to the BTP_1 to generate the *user_hash* parameter (eq.4). Finally, the *proof* is the encryption (E) of the *user_key* combined with the *user_hash*, using BTP_1 as the key (eq.5).

$$\text{hash\_function}(BTP_1) \ = \ user\_hash \tag{4}$$

$$proof \ = \ E(BTP_1, user\_key \ + \ user\_hash) \tag{5}$$

The Authentication server stores *username*, *opt_key*, *biometric_key*, *user_hash*, *user_key*, and *proof*, and is unaware of the biometric characteristics, which prevents it from decrypting the *proof*. The Client stores the *username*, *opt_key* and *biometric_key* in a successful registration. Finally, all parameters are discarded on the Registration server.

3.3 Validation Phase

The validation phase aims to authenticate a legitimate Client who has been previously registered. In a secure communication, the Client sends their *username* and a new facial image F_2, which differs from the previously submitted, F_1. Subsequently, the authentication server retrieves the corresponding *user_key* and *otp_key* from the database using the provided *username*. The server then generates a Hash based Message Authentication Code (HMAC) by applying an HMAC function that uses the *user_key* along with a randomly generated string produced by a PRNG function: MAC $=$ HMAC(*user_key*, PRNG()).

The MAC code is encrypted (E) using disposable codes generated by the TOTP function, which is initialized with the client's *otp_key* (eq. 6), producing the cryptographic object P_1 (eq. 7). The TOTP function creates an authentication mechanism using temporary *unique_codes* that are valid for a short period, specifically 30 seconds.

$$\text{TOTP}(otp\_key) \ = \ unique\_code_i \tag{6}$$

$$\text{E}(unique\_code_i, \text{MAC}) \ = \ P_1 \tag{7}$$

The Client begins by decrypting (D) the object P_1 with the disposable code (eq. 8) obtained from the TOTP function initialized with the *otp_key* (eq. 9). Next, the CNN model is used to process two photographs, the registration photo F_1 and the validation photo F_2, to obtain their respective embeddings BT_1 (eq. 10) and BT_2 (eq. 11). The *delta* is then calculated as the distance between these two embeddings (eq. 12). Our transformation function BTP initialized with the biometric_key is applied to BT_2, canceling out the biometric characteristics and producing BTP_2 (eq. 13). Subsequently, the Client encrypts (E) the MAC with the BTP_2, producing the encrypted object P_2 (eq. 14). P_2 and *delta* are subsequently transmitted to the Authentication server.

$$\text{D}(unique\_code_i, \ P_1) \ = \ \text{MAC} \tag{8}$$

$$\text{TOTP}(otp\_key) \ = \ unique\_code_i \tag{9}$$

$$\text{CNN}(F_1) \ = \ BT_1 \tag{10}$$

$$\text{CNN}(F_2) \ = \ BT_2 \tag{11}$$

$$\text{dist}(BT_1, BT_2) \ = \ delta \tag{12}$$

$$\text{BTP}(biometric\_key, BT_2) \ = \ BTP_2 \tag{13}$$

$$\text{E}(BTP_2, \ \text{MAC}) \ = \ P_2 \tag{14}$$

The Authentication server uses the CNN model to generate BT_2' from F_2, previously received (eq. 15). The biometric features BT_2' are transformed using our BTP function initialized with the *biometric_key*, producing BTP_2' (eq. 16). The server decrypts (D) P_2 with BTP_2' (eq. 17) and validates if this result matches the MAC constructed previously (eq. 18). If it is not possible to obtain the exact same MAC, then the procedure is terminated as unauthorized. Next, the server obtains the biometric characteristics BT_1' by applying the $delta$ to the vector BT_2' (eq. 19). To produce BTP_1', the biometric characteristics BT_1' are canceled using the BTP function initialized with the $biometric\_key$ (eq. 20). The server decrypts (D) the $proof$ with the BTP_1' (eq. 21) and tests if the result is equal to the $user\_key$ combined with the $user\_hash$. Finally, the procedure is concluded as authorized if the similarity between BT_1' and BT_2' is above a certain threshold (eq. 22).

Upon successful validation, the Authentication server acquires the biometric characteristics BT_1 and BT_2 for facial recognition purposes, which are not stored in the server database. By utilizing distinct keys, it is feasible to generate new and different $proofs$ for the same image F_1.

This approach was designed to support various hashing and encryption methods; thus, the selected cryptographic functions and their implementation will be discussed in Subsection 3.4.

$$\text{CNN}(F_2) = BT_2' \quad (15)$$
$$\text{BTP}(biometric\_key, BT_2') = BTP_2' \quad (16)$$
$$\text{D}(BTP_2', P_2) = \text{MAC}' \quad (17)$$
$$\text{MAC} \iff \text{MAC}' \quad (18)$$
$$\text{dist}(BT_2', delta) = BT_1' \quad (19)$$
$$\text{BTP}(biometric\_key, BT_1') = BTP_1' \quad (20)$$
$$\text{D}(BTP_1', proof) == user\_key' + user\_hash' \quad (21)$$
$$\text{similarity}(BT_1', BT_2') > threshold \quad (22)$$

3.4 Implementation details

The approach presented was implemented using Python and evaluated on a server running Ubuntu 20.04, equipped with an AMD Ryzen 7 5700G processor and 58GB of RAM. As PRNG, we utilized the *SystemRandom* function to produce the necessary random parameters. For symmetric encryption, we chose AES-256, which is available through the *cryptography* library. Additionally, we utilized SHA-512 as hashing function, imported from the *hashlib* library. The BTP function, as previously detailed in Subsection 3.2, was developed by our team, using the strategy of [Hahn and Marcel, 2022b].

Biometric features are typically evaluated through a similarity function, as demonstrated in prior studies. In our proposed approach, however, we introduce deformations to these data. In Section 4, we present the impact of this approach on the effectiveness of facial authentication.

4 CRITICAL ANALYSIS

This section outlines the evaluation plan for the proposed approach, detailing the benchmarks and selected datasets, as well as presenting the results of the experiments conducted in subsection 4.1. In subsection 4.2, a critical assessment of the method is provided, justifying the design choices made and discussing the advantages and disadvantages that these choices impose on FRS.

4.1 Experiments and Results

The Subsection is structured into datasets, CNN model, and results.

Datasets. The criterion for selecting the data was the preference for high-quality datasets with minimal noise, along with the availability. Therefore, this work utilizes models trained on the MSIM-V2 dataset [Deng et al., 2019], known for its lower noise levels compared to datasets such as the MS-Celeb-1M dataset [Guo et al., 2016]. The datasets used for validation include LFW [Huang et al., 2008], AgeDB-30 [Moschoglou et al., 2017], CALFW [Zheng et al., 2017], CPLFW [Zheng and Deng, 2018], RFW [Wang et al., 2019], and XQLFW [Knoche et al., 2021]. These validation datasets present a challenge as they consist of in-the-wild data, where many frames may not meet quality standards (such as blur, pixelation, or closed eyes). The images are 112x112 in size and are aligned with the guidelines set forth in ArcFace [Deng et al., 2019]. Each dataset comprises 6000 test cases.

CNN model. The MagFace model was selected because, based on our current knowledge, it consistently produces results that are at the forefront of the field. The neural network was instantiated using the checkpoint files provided by the authors of MagFace after training with the MSIM-V2 dataset, employing stochastic gradient descent as the optimization algorithm [Meng et al., 2021]. The cosine distance was utilized as the similarity metric for comparing the feature embeddings.

Results. The results obtained for the AgeDB-30, CALFW, LFW, RFW (African, Caucasian, Indian), and XQLFW benchmarks are presented in Table 3. Our approach performed well on most benchmarks in terms of accuracy, with the highest accuracy achieved on LFW at 99.43%, followed closely by AgeDB-30 and RFW African/Caucasian/Indian at 98%. The Equal Error Rate (EER) values were relatively low for

Table 3: Our approach evaluated in different benchmarks by the following metrics: accuracy, standard deviation (std), EER, and AUC.

Dataset	Accuracy +-std	EER	AUC
AGEDB30	0.981 +-0.009	0.02320	0.99176
CALFW	0.958 +-0.036	0.52459	0.42706
LFW	0.994 +-0.004	0.00624	0.99647
African	0.986 +-0.004	0.01587	0.99323
Asian	0.976 +-0.006	0.02358	0.99474
Caucasian	0.989 +-0.003	0.01289	0.99515
Indian	0.982 +-0.007	0.02032	0.99436
XQLFW	0.837 +-0.019	0.17128	0.90987

most datasets, indicating good performance in terms of false match rate and false non-match rate. The LFW dataset had the lowest EER at 0.62%, while CALFW and XQLFW had the highest, with rates of 52% and 17%, respectively. The AUC values were also high on most datasets, with LFW again leading with an AUC of 99.65%. The other datasets, including AgeDB-30, RFW African, RFW Asian, RFW Caucasian, and RFW Indian, also showed high AUC values above 99%. Overall, the findings suggest that our approach excelled on the benchmark datasets, particularly on LFW, demonstrating high accuracy and discriminative performance across various demographic groups. The ROC curve in Fig. 2 further illustrates the system's performance. Although the proposed technique maintains a high level of accuracy without compromising overall performance, future research efforts should focus on enhancing performance on datasets with lower accuracy and higher Equal Error Rate (EER) values, such as the XQLFW dataset. However, it is important to note that tests on low-quality datasets should be approached with caution. Further investigation is necessary to improve performance in others benchmarks.

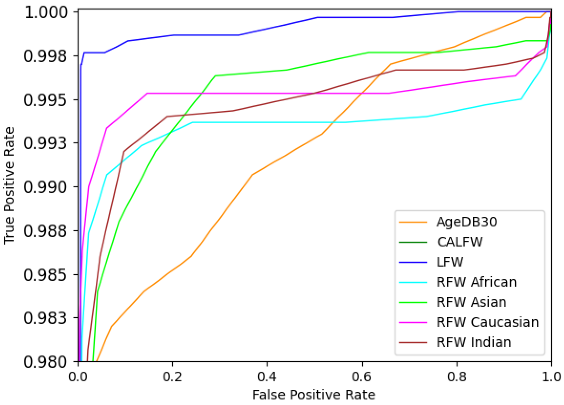

Figure 2: ROC curve evaluating our approach across different benchmarks.

Table 4 presents the True Acceptance Rate (TAR) at a False Acceptance Rate (FAR) ranging from 0.0001 to 0.1 for various validation datasets. The AgeDB-30 dataset demonstrates a high TAR of 98.94%, whereas the CALFW dataset shows a TAR of 94.17% at a FAR of 0.0001. The CPLFW dataset initially exhibits a lower TAR of 28.95%, which increases substantially to 98.25% at a FAR of 0.01. On the other hand, the LFW dataset starts with a modest TAR of 28.3% at a FAR of 0.0001 but improves significantly to 99.91% at a FAR of 0.01. The RFW dataset generally displays high TAR values at a FAR of 0.0001, ranging from 97.47% to 99.75%. In comparison, the XQLFW initially shows a TAR of 52.10% at a FAR of 0.0001 but increases as the FAR values rise. The diverse performance of face recognition systems across different datasets is highlighted, with some datasets achieving high TAR even at low FAR values, while others exhibit substantial enhancements as FAR increases.

Table 4: Accuracy of our approach (%) presenting TAR in FARs ranging from 0.0001 to 0.1 across different benchmarks.

Validation	TAR@FAR			
Datasets	0.0001	0.001	0.01	0.1
AGEDB30	98.94	99.49	99.49	99.53
CALFW	94.17	98.99	98.99	98.99
CPLFW	28.95	97.78	98.25	98.25
LFW	28.3	53.19	99.91	99.91
RFW African	99.31	99.79	99.88	99.91
RFW Asian	97.47	99.16	99.88	99.93
RFW Caucasian	99.75	99.89	99.94	99.94
RFW Indian	99.22	99.73	99.84	99.84
XQLFW	52.10	74.42	75.19	86.18

The results demonstrate that our approach achieves performance levels that are comparable to the current state of the art. When compared to MagFace [Meng et al., 2021] as a baseline, our method exhibits minimal differences of 0.08, 1.36, 2.84, and 5.38 on LFW, AgeDB-30, CALFW, and CPLFW datasets, respectively (see Table 5). Our design integrates various protective strategies for facial recognition while facilitating accurate decision-making through a systematic comparison process.

Table 5: Verification accuracy (%) on the LFW, AgeDB-30, CALFW, and CPLFW benchmarks for the ArcFace [Deng et al., 2019], MagFace [Meng et al., 2021], and our proposed method.

Datasets	ArcFace	MagFace	Our	Gain
LFW	99.81	99.83	99.91	0.08
AgeDB30	98.05	98.17	99.53	1.36
CALFW	95.96	96.15	98.99	2.84
CPLFW	92.72	92.87	98.25	5.38

4.2 Critical Evaluation

The registration server operates as a Trusted Third-Party (TTP), managing temporary and revocable biometric records. To ensure its trustworthiness, it is essential that our approach is both open-source and transparent. This transparency facilitates analysis, enhancement, and the detection of vulnerabilities.

The advantage of this approach lies in the fact that the database does not store biometric data. As a result, it is not possible to recognize users unless they can be linked to the parameters *username*, *user_key*, or *biometric_key*. To mitigate this risk, it is essential to adopt best practices by generating random parameters for these attributes. This way, identifying clients through the database becomes challenging or even impossible without additional sources of information, such as network traffic captures or system logs.

Brute force attacks on servers can be mitigated through the implementation of Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA), which are designed to differentiate between human users and automated algorithms. We recommend hCaptcha[2] due to its strong emphasis on user privacy, which can be particularly advantageous for companies that prioritize compliance with data protection regulations such as the General Data Protection Regulation (GDPR). Additionally, protection against Distributed Denial-of-Service (DDoS) attacks is enhanced by leveraging an external service, thereby preventing the extensive exploitation of computational resources associated with FRS.

In FRS, it is crucial to achieve a balance between the accuracy of authentication and the protection of biometric data privacy. A drawback of the current approach is that the authentication server is required to compute the biometric features of the client, referred to as BT_1 and BT_2. This necessity creates a compromise between the client's privacy and the application's functionality, facilitating client authentication through facial recognition. The complexities of this authentication process revolve around the server's ability to determine whether $delta$ and BT_2 are derived from the same client data represented by BT_1. This verification relies on the assumption that BT_1 and BT_2 exhibit similar statistical distributions and that they diverge from one another by a defined parameter, $delta$. As a result, the system aims to achieve reliable authentication while simultaneously safeguarding the privacy of the biometric data stored in its persistent database.

To successfully impersonate a user, an attacker would need to obtain the user's registration photograph F_1, a second validation photograph F_2, the *username*, the opt_key, the *biometric_key*, and the corresponding CNN model to generate BT_1, BT_2, $delta$, and BTP_2. This could be achievable if the attacker gains unauthorized access to the client's device or the authentication server. A Man-in-the-Middle (MITM) position would only be feasible if the communications were not conducted securely, for instance, without the use of Transport Layer Security.

The proposed solution has several drawbacks that suggest areas for future research. First, the registration photo is stored on the client side, meaning that losing it would necessitate a new photo submission for biometric registration. Second, compliance with guidelines such as the ICAO standard is crucial to ensure the photo meets biometric requirements. Third, both F_1 and F_2 are static objects; if an attacker accesses the client device, these could be exploited for spoofing.

As a future direction, the solution could incorporate Active Liveness Detection, utilizing dynamic video and audio inputs to enhance security against spoofing. Lastly, refining the registration process to better align with ICAO standards could improve reliability.

5 CONCLUSIONS

In conclusion, our approach to MFA for facial recognition technology prioritizes the privacy of biometric data by not storing biometric data. Instead, it employs cryptographic algorithms to generate signatures (proofs) confirming the user's identity. The results indicate excellent performance in face recognition tasks across various benchmark datasets, demonstrating high accuracy and AUC values, particularly on the LFW dataset. The TAR at different FAR levels further underscores the reliability of the method, placing it on par with state-of-the-art solutions, albeit with slight variances in performance metrics.

Overall, our approach excels by integrating multiple techniques to enhance precision and efficacy of FRS, paving the way for advanced applications across numerous domains. Notably, it avoids the storage of biometric data and its representations within a database; the generated proofs do not contain any biometric information and can be revoked or recreated by simply changing the key. Additionally, the use of standard algorithms and hash functions facilitates the creation of robust proofs, leveraging flexibility in combination with other CNN and cryptographic algorithms.

[2] https://www.hcaptcha.com

Nevertheless, certain limitations persist, as security depends on the careful validation of data submitted by the client. Future work could focus on incorporating methods for image assessment and live detection techniques to further enhance the security of our approach.

6 ACKNOWLEDGEMENTS

This work has been supported by the Fundação para a Ciência e a Tecnologia (FCT) and the Fundo Social Europeu (FSE) through research grant 2023.02790.BD and by project UIDB/00048/20202. The authors would like to thank the INCM, ISR-UC, and CISUC. The computational part of this study was supported by VISTeam at ISR-Coimbra.

REFERENCES

Abdullahi, S. M., Sun, S., Wang, B., Wei, N., and Wang, H. (2024). Biometric template attacks and recent protection mechanisms: A survey. Information Fusion, 103:102144.

Bahmanisangesari, S. (2024). A novel transformer-based multi-step approach for predicting common vulnerability severity score.

Boddeti, V. N. (2018). Secure face matching using fully homomorphic encryption. In 2018 IEEE 9th international conference on biometrics theory, applications and systems (BTAS), pages 1–10. IEEE.

Dabbah, M., Woo, W., and Dlay, S. (2007). Secure authentication for face recognition. In 2007 IEEE symposium on computational intelligence in image and signal processing, pages 121–126. IEEE.

Dang, T. and Choi, D. (2020). A survey on face-based cryptographic key generation. Smart Media Journal, 9(2):39–50.

Delac, K. and Grgic, M. (2004). A survey of biometric recognition methods. In Proceedings. Elmar-2004. 46th International Symposium on Electronics in Marine, pages 184–193. IEEE.

Deng, J., Guo, J., Xue, N., and Zafeiriou, S. (2019). 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.

Dong, X., Wong, K., Jin, Z., and Dugelay, J.-l. (2019). A cancellable face template scheme based on nonlinear multi-dimension spectral hashing. In 2019 7th International Workshop on Biometrics and Forensics (IWBF), pages 1–6. IEEE.

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

Drozdowski, P., Stockhardt, F., Rathgeb, C., Osorio-Roig, D., and Busch, C. (2021a). Feature fusion methods for indexing and retrieval of biometric data: Application to face recognition with privacy protection. IEEE Access, 9:139361–139378.

Drozdowski, P., Stockhardt, F., Rathgeb, C., Osorio-Roig, D., and Busch, C. (2021b). Feature fusion methods for indexing and retrieval of biometric data: Application to face recognition with privacy protection. IEEE Access, 9:139361–139378.

Ekka, M. U., Mze, O. A., Singh, T., and Raghava, N. (2022). Attendance management system using modern face recognition and gesture recognition using deep learning. In Advances in Manufacturing Technology and Management: Proceedings of 6th International Conference on Advanced Production and Industrial Engineering (ICAPIE)—2021, pages 527–535. Springer.

El-Shafai, W., Mohamed, F. A. H. E., Elkamchouchi, H. M., Abd-Elnaby, M., and Elshafee, A. (2021). Efficient and secure cancelable biometric authentication framework based on genetic encryption algorithm. IEEE Access, 9:77675–77692.

Engelsma, J. J., Jain, A. K., and Boddeti, V. N. (2022). Hers: Homomorphically encrypted representation search. IEEE Transactions on Biometrics, Behavior, and Identity Science, 4(3):349–360.

Gioacchini, L., Mellia, M., Drago, I., Delsanto, A., Siracusano, G., and Bifulco, R. (2024). Autopenbench: Benchmarking generative agents for penetration testing. arXiv preprint arXiv:2410.03225.

Guo, Y., Zhang, L., Hu, Y., He, X., and Gao, J. (2016). Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14, pages 87–102. Springer.

Hahn, V. K. and Marcel, S. (2022a). Biometric template protection for neural-network-based face recognition systems: A survey of methods and evaluation techniques. IEEE Transactions on Information Forensics and Security, 18:639–666.

Hahn, V. K. and Marcel, S. (2022b). Towards protecting face embeddings in mobile face verification scenarios. IEEE Transactions on Biometrics, Behavior, and Identity Science, 4(1):117–134.

Huang, G. B., Mattar, M., Berg, T., and Learned-Miller, E. (2008). 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.

Jindal, A. K., Shaik, I., Vasudha, V., Chalamala, S. R., Ma, R., and Lodha, S. (2020). 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. IEEE.

Juels, A. and Wattenberg, M. (1999). A fuzzy commitment scheme. In Proceedings of the 6th ACM conference on Computer and communications security, pages 28–36.

Keller, D., Osadchy, M., and Dunkelman, O. (2020). Fuzzy commitments offer insufficient protection to biometric templates produced by deep learning. arXiv preprint arXiv:2012.13293.

Keller, D., Osadchy, M., and Dunkelman, O. (2021). Inverting binarizations of facial templates produced by deep learning (and its implications). IEEE Transactions on Information Forensics and Security, 16:4184–4196.

Kirchgasser, S., Uhl, A., Martinez-Diaz, Y., and Mendez-Vazquez, H. (2020). Is warping-based cancellable biometrics (still) sensible for face recognition? In 2020 IEEE International joint conference on biometrics (IJCB), pages 1–9. IEEE.

Knoche, M., Hormann, S., and Rigoll, G. (2021). Cross-quality lfw: A database for analyzing cross-resolution image face recognition in unconstrained environments. In 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021), pages 1–5. IEEE.

Lala, S. K., Kumar, A., and Subbulakshmi, T. (2021). Secure web development using owasp guidelines. In 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), pages 323–332. IEEE.

Li, S. and Kot, A. C. (2010). Privacy protection of fingerprint database. IEEE Signal Processing Letters, 18(2):115–118.

Li, S. and Kot, A. C. (2012). Fingerprint combination for privacy protection. IEEE transactions on information forensics and security, 8(2):350–360.

Ma, Y., Wu, L., Gu, X., He, J., and Yang, Z. (2017). A secure face-verification scheme based on homomorphic encryption and deep neural networks. IEEE Access, 5:16532–16538.

Manisha and Kumar, N. (2020). On generating cancelable biometric templates using visual secret sharing. In Arai, K., Kapoor, S., and Bhatia, R., editors, Intelligent Computing, pages 532–544, Cham. Springer International Publishing.

Marcel, S., Fierrez, J., and Evans, N. (2023). Handbook of Biometric Anti-Spoofing: Presentation Attack Detection and Vulnerability Assessment, volume 1. Springer.

Meng, Q., Zhao, S., Huang, Z., and Zhou, F. (2021). 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.

Moschoglou, S., Papaioannou, A., Sagonas, C., Deng, J., Kotsia, I., and Zafeiriou, S. (2017). 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.

Mousa, A., Karabatak, M., and Mustafa, T. (2020). Database security threats and challenges. In 2020 8th International Symposium on Digital Forensics and Security (ISDFS), pages 1–5. IEEE.

National Inst Of Standards And Technology Gaithersburg Md (2001). Advanced Encryption Standard (AES). https://apps.dtic.mil/sti/citations/ADA403903. (accessed: April, 2024).

Osorio-Roig, D., Rathgeb, C., Drozdowski, P., and Busch, C. (2021). Stable hash generation for efficient privacy-preserving face identification. IEEE Transactions on Biometrics, Behavior, and Identity Science, 4(3):333–348.

Ramu, T. and Arivoli, T. (2012). Biometric template security: an overview. In Proceedings of International Conference on Electronics, volume 65.

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

Rui, Z. and Yan, Z. (2019). A survey on biometric authentication: Toward secure and privacy-preserving identification. IEEE Access, 7:5994–6009.

Sarkar, A. and Singh, B. K. (2020). A review on performance, security and various biometric template protection schemes for biometric authentication systems. Multimedia Tools and Applications, 79(37):27721–27776.

Schroff, F., Kalenichenko, D., and Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 815–823.

Singh, A., Srivastva, R., and Singh, Y. N. (2021). Plexnet: An ensemble of deep neural networks for biometric template protection. International Journal of Advanced Computer Science and Applications, 12(4).

Smith, R. and Xu, J. (2011). A survey of personal privacy protection in public service mashups. In Proceedings of 2011 IEEE 6th International Symposium on Service Oriented System (SOSE), pages 214–224. IEEE.

Tremoço, J., Medvedev, I., and Gonçalves, N. (2021). Qualface: Adapting deep learning face recognition for id and travel documents with quality assessment. In 2021 International Conference of the Biometrics Special Interest Group (BIOSIG), pages 1–6.

Wang, M., Deng, W., Hu, J., Tao, X., and Huang, Y. (2019). Racial faces in the wild: Reducing racial bias by information maximization adaptation network. In Proceedings of the ieee/cvf international conference on computer vision, pages 692–702.

Wu, H., Wu, J., Wu, R., Sharma, A., Machiry, A., and Bianchi, A. Veribin: Adaptive verification of patches at the binary level.

Zheng, T. and Deng, W. (2018). Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments. Beijing University of Posts and Telecommunications, Tech. Rep, 5(7):5.

Zheng, T., Deng, W., and Hu, J. (2017). Cross-age lfw: A database for studying cross-age face recognition in unconstrained environments. arXiv preprint arXiv:1708.08197.