Enhancing Privacy and Security in AI using Fully Homomorphic Encryption

by

Arjun Ramesh Kaushik

May 15 2024

A dissertation submitted to the

Faculty of the Graduate School of

the University at Buffalo, The State University of New York

in partial fulfilment of the requirements for the

degree of

Master of Science

Department of Computer Science and Engineering

To my family and friends

Acknowledgments

I extend my sincere gratitude to the Computer Science Department of University at Buffalo, SUNY and Dr. Nalini Ratha for providing me with an opportunity to pursue my research interests in the intersection of Computer Vision and Privacy. Special appreciation goes to my lab mates – Bharat Yalavarthi, Tilak Sharma and Vatsal Aggarwal - for their timely help and inputs.

Table of Contents

List of Tables

List of Figures

Abstract

Artificial intelligence (AI) systems have permeated everyday life and business landscapes, serving as vital aids in human decision-making processes. Their evolution has led to heightened complexity and efficacy, with applicability across diverse domains. However, widespread acceptance and utilization of AI hinge on the establishment of trust in their outcomes—a concept encapsulated by the term "Trustworthy AI". In this research, we explore the privacy and security facets of Trustworthy AI. We propose novel frameworks mitigating privacy and security risks in face recognition systems and Unmanned Aerial Vehicles (UAVs) using Fully Homomorphic Encryption (FHE).

Modern face recognition systems utilize deep neural networks to extract salient features from a face. These features denote embeddings in latent space and are often susceptible to data leakage and, in some cases, can even be used to reconstruct the original face image. To prevent compromising identities, template protection schemes are commonly employed. However, these schemes may still not prevent the leakage of soft biometric information such as age, gender and race. To alleviate this issue, we propose a novel technique that combines FHE with an existing template protection scheme known as PolyProtect. Our proposed approach ensures irreversibility and unlinkability, effectively preventing the leakage of soft biometric attributes from face embeddings without compromising recognition accuracy.

Autonomous Unmanned Aerial Vehicles (UAVs) have become essential tools in defense, law enforcement, disaster response, and product delivery. These autonomous navigation systems require a wireless communication network, and of late are deep learning based. In critical scenarios such as border protection or disaster response, ensuring the secure navigation of autonomous UAVs is paramount. But, these autonomous UAVs are susceptible to adversarial attacks through the communication network or the deep learning models - eavesdropping / man-in-the-middle / membership inference / reconstruction. Time is of

essence in such critical situations, hence, we propose an FHE-optimized model using Knowledge Distillation for compression. Our compressed model showcases an 18x improvement in time for secure autonomous UAV navigation. Additionally, we demonstrate the efficacy of our proposed approach through extensive experimentation. Our proposed approach ensures feasible, secure and private autonomous UAV navigation with negligible loss in performance.

Collectively, these contributions underscore the advancement of Trustworthy AI, addressing critical challenges in privacy and security, thereby paving the way for the deployment of AI technologies in sensitive domains with enhanced reliability and resilience.

Chapter1

Introduction

Machine Learning (ML) can be broadly defined as empowering machines with large datasets to analyze and make informed decisions. As ML is integrated into real-world products and services, it encounters significant challenges. Models may struggle to adapt to slight variations in data distribution, inadvertently incorporate sensitive features leading to unfair treatment of certain demographic groups, and lack transparency in explaining decisions to end-users, such as medical professionals. These issues collectively contribute to a lack of trust in current ML technologies. A considerable portion of current ML research is dedicated to advancing Trustworthy ML. This report delves into the privacy and security facets of Trustworthy ML.

With machines consuming such vast data, privacy and security of users is a necessity. Privacy and Security in ML can achieved through numerous techniques - Differential Privacy, Secure Multiparty Computation, Data Anonymization, Federated Learning and Homomorphic Encryption. We focus on a type of Homomorphic Encryption, namely, Fully Homomorphic Encryption (FHE). We elicit on the efficacy of FHE in bolstering privacy and security through two applications in the real world as shown in Fig. 1.2 - i) **Face analytics** : Protecting soft-biometric attributes from face embeddings while preserving identification accuracy; ii) **Autonomous UAVs** : Private and secure navigation in autonomous UAVs.

Face Analytics. Face recognition entails the extraction of features from face images and comparing them to either validate a claimed identity ("verification") or determine an identity

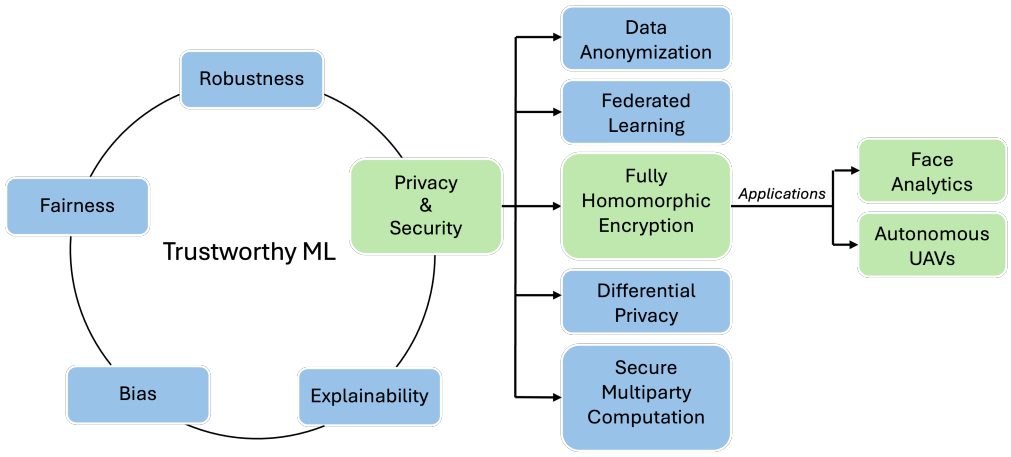


Figure 1.1: This report focuses on furthering Trustworthy Machine Learning research by leveraging Fully Homomorphic Encryption (FHE) to bolster privacy and security. It showcases FHE's efficacy through real-life applications in face analytics and autonomous UAVs.

("identification") [1]. Recent advancements in deep neural networks and AI have resulted in the development of powerful face recognition systems [2, 3, 4] that can be deployed in a wide range of applications such as personalized services, law enforcement, border security, and smartphone access [5]. However, this development has also raised questions about privacy accorded to subjects and the security of the templates (such as embeddings) stored in a face recognition application [6]. Even as ethical concerns attendant to the technology are being rightfully discussed in public forums, it is necessary for the technology itself, on the one hand, and its users, on the other hand, to embrace measures that can enhance privacy and security while mitigating potential biases [7]. Otherwise, the technology runs the risk of being overwhelmed by restrictive legislation [8, 9] that can stifle the benefits of this technology in solving egregious crimes [10].

To address some of the privacy challenges associated with *face embeddings* stored as *templates*, we propose an approach in this paper that employs a polynomial transformation on homomorphically encrypted face embeddings. Using Fully Homomorphic Encryption (FHE) in our proposed method ensures that the face recognition result is only disclosed to authorized parties with the secret key for homomorphic encryption, and the face embeddings themselves are secured through encryption during the recognition process. We show through

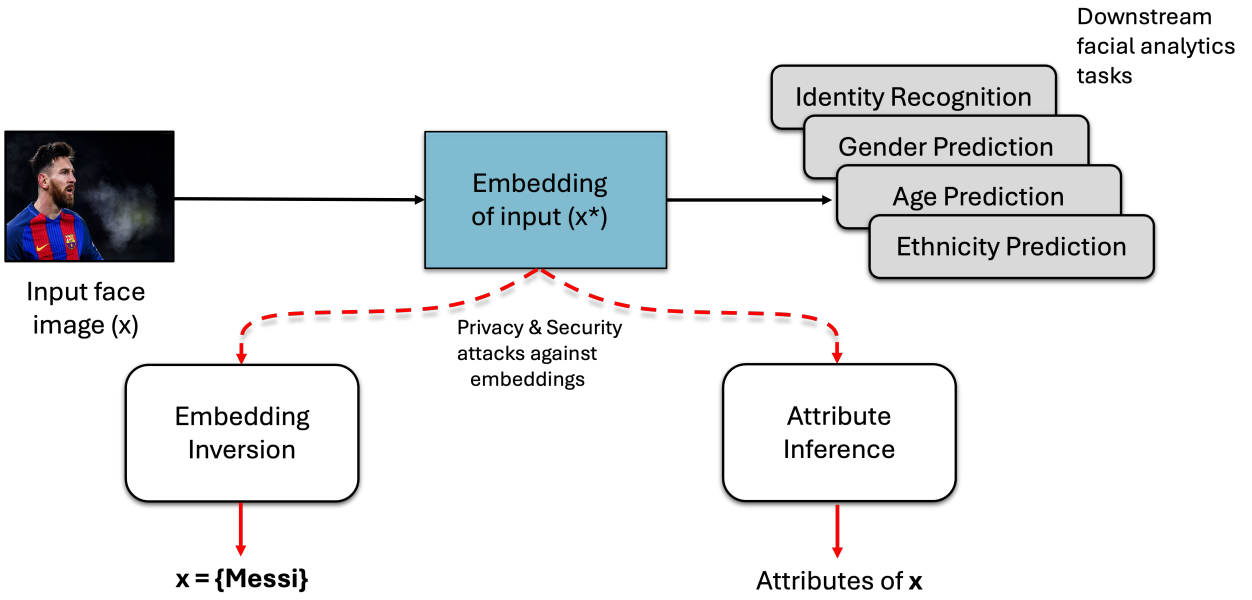


Figure 1.2: Taxonomy of attacks against face embeddings. We assume the adversary has access to the face embeddings (x*) that will be used for facial analysis tasks, and explore different protection techniques against these attacks.

our experiments that using FHE also prevents leakage of soft biometrics (e.g., age, gender/-sex, race/ethnicity) from face embeddings. (It is necessary to point out that there is a difference between *race* and *ethnicity*, as well as *sex* and *gender*. In this report, however, we use these terms interchangeably).

Autonomous UAVs. Unmanned Aerial Vehicles (UAVs), commonly referred to as drones, are defined as aircrafts that operate without any human onboard. UAVs have brought about transformative changes across various industries, providing unmatched capabilities in surveillance, reconnaissance, disaster response, and product delivery [11]. As the demand for more complex tasks performed by UAVs grows, so do the challenges in their development, particularly in striving for fully autonomous operation with minimal human intervention. Effective deployment of autonomous UAVs requires intricate path planning, obstacle detection, intelligent maneuverability and a wireless communication network. Research efforts on autonomous navigation in UAVs for visual mapping, obstacle detection, and path planning have gravitated towards deep neural networks [12, 13, 14, 15]. In critical scenarios such as

surveillance and disaster response, a secure wireless communication network to ensure secure navigation is imperative. In addition to susceptible wireless networks, deep neural networks in drones are also vulnerable to adversarial attacks [16, 17].

Previous works have explored computer vision-based autonomous UAV systems [12], whereas, recent efforts take a Reinforcement Learning (RL) approach [13, 14, 15]. In our work, we adopt the Actor-Critic model with Proximal Policy Optimization (PPO) as the policy gradient algorithm to demonstrate a solution addressing the privacy and security challenges in autonomous UAVs. While proposing an end-to-end framework merging RL and FHE for secure inferencing on encrypted inputs, [18] overlook addressing the high latency issue in their model. In our work, we enhance the inference speed by 18x of the model proposed in [18]. We achieve this through model compression using Knowledge Distillation in 2 steps. We show through our experiments that, with FHE, the navigation results are unaffected on the compressed model while guaranteeing utmost security and low latency.

Chapter2

Fully Homomorphic Encryption

2.1 Homomorphic Encryption

Encryption is the process where plain-text data is encrypted into ciphertext using a secret key and a cryptographic algorithm. Only authorized entities with a private key can decrypt the ciphertext back to the plaintext. Encryption is essential for protecting sensitive data from unauthorized access or modification. Homomorphic Encryption (HE) is a cryptographic

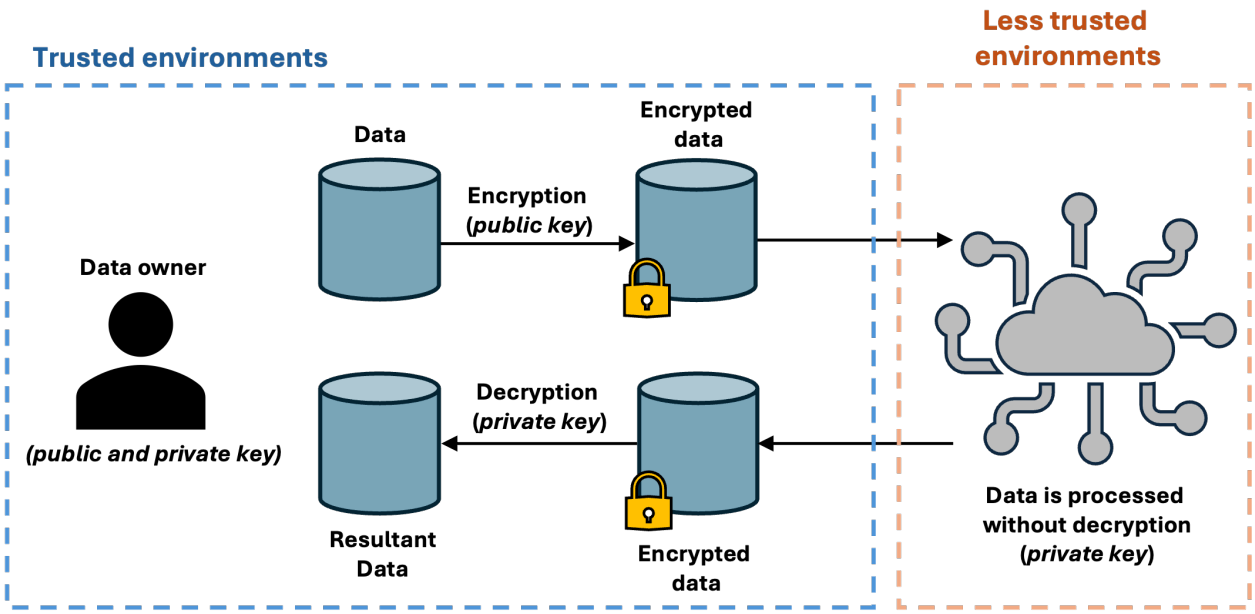


Figure 2.1: An overview of the workings of a homomorphic system.

system that permits certain computations to be performed on encrypted data without re-

quiring decryption [19] as shown in Fig. 2.1. In this system, we have public (pk) and secret (sk) keys, encryption (E) and decryption (D) mechanisms, and plaintext values x and y. When x and y are encrypted as $x' = E(x, pk)$ and $y' = E(y, pk)$, respectively, a cryptosystem is considered homomorphic with respect to a chosen operator (e.g., addition or multiplication), denoted as $\circ$, if we can find another operator $\bullet$ such that $x \circ y = D(x' \bullet y', sk)$. This means that we can conduct operations on encrypted data and obtain the same result when decrypting using the private secret key.

Specifically, given $c_i = E(x_i, pk), i = 1, 2, \cdots, K$, an FHE scheme allows the computation of $c = g(c_1, c_2, \cdots, c_K)$ such that $D(c, sk) = f(x_1, x_2, \cdots, x_K)$ for any arbitrary function f.

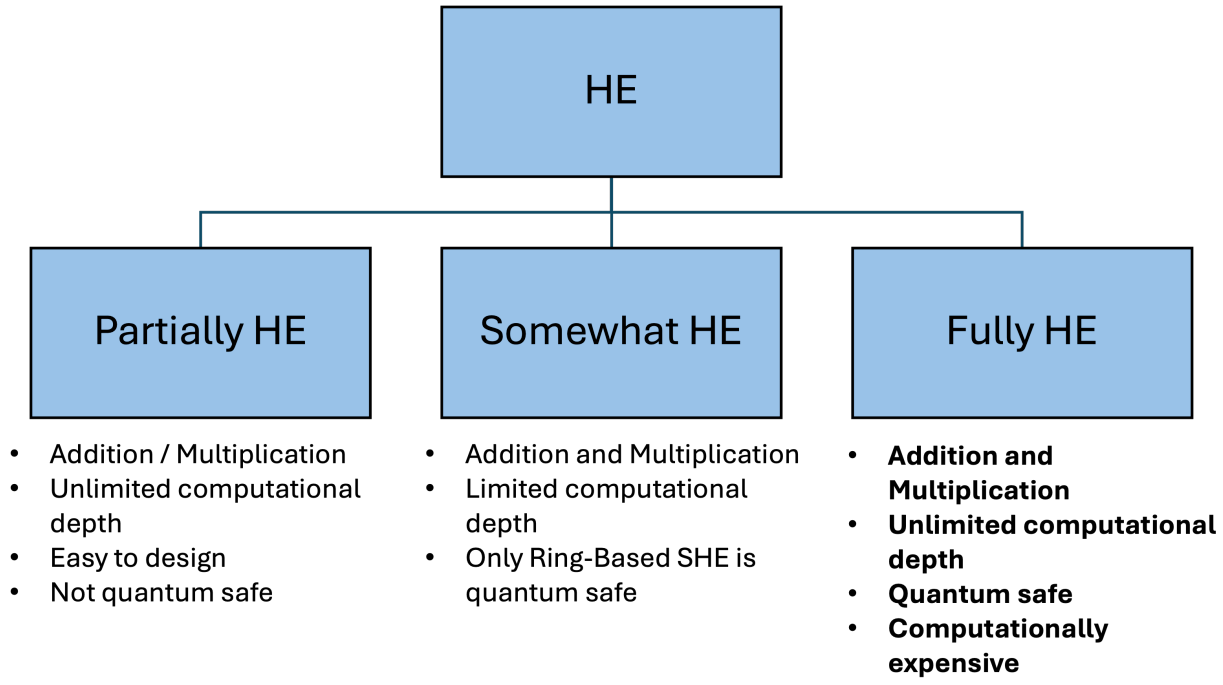


Figure 2.2: Types of Homomorphic Encryption (HE) and their characteristics.

It is essential to note that there are three types of homomorphic encryption schemes [20] as shown in Fig. 2.2:

- **Partial Homomorphic Encryption (PHE)** permits addition or multiplication operations.

- **Somewhat Homomorphic Encryption (SHE)** allows limited computations on ciphertexts.
- **Fully Homomorphic Encryption (FHE)** enables computations on ciphertexts of any depth and complexity.

2.2 Fully Homomorphic Encryption

Fully Homomorphic Encryption (FHE) allows computations of unlimited depth and is quantum safe. Data remains secure and private in untrusted environments, like public clouds or external parties. The data stays encrypted at all times, which minimizes the likelihood that sensitive information ever gets compromised. On the other hand, FHE remains commercially infeasible for computationally-heavy applications, as of today.

Numerous FHE systems have been introduced, including the Brakerski/Fan-Vercauteren (BFV), Brakerski-Gentry-Vaikuntanathan (BGV), and Cheon-Kim-Kim-Song (CKKS) schemes [21]. The BFV and BGV schemes enable vector operations involving integers, while the CKKS scheme facilitates floating-point operations. These schemes achieve Single Instruction Multiple Data (SIMD) operations by bundling plaintext values into an array and then encrypting them to get ciphertext.

2.3 FHE Libraries

There are several popular open-source FHE libraries available, each with its own strengths and trade-offs. Here are some of the notable ones and how they compare:

2.3.1 HElib

HElib [22] is one of the earliest and most widely used FHE libraries, developed by researchers at IBM. It implements the Brakerski-Gentry-Vaikuntanathan (BGV) scheme and supports

efficient SIMD operations. Key features:

- Supports arbitrary depth computations on encrypted data.
- Efficient for SIMD operations and data-parallel applications.
- Actively maintained and well-documented Written in C++.

2.3.2 SEAL

SEAL (Simple Encrypted Arithmetic Library) [23] is a Microsoft-developed library that implements the Brakerski/Fan-Vercauteren (BFV) scheme. Its key highlights are:

- Supports computations up to a specific depth.
- Optimized for large-scale computations.
- Easy to use API.
- Written in C++.

2.3.3 PALISADE

PALISADE (previously called OpenFHE) [24] is a modular library that incorporates several FHE schemes like BGV, BFV, and CKKS. It is designed to be extensible and support different use cases. Features:

- Supports multiple FHE schemes and SIMD/packed operations.
- Modular design for easy integration of new schemes.
- Active development and community engagement.
- Written in C++.

2.3.4 TFHE

TFHE (Fast Fully Homomorphic Encryption over the Torus) [25] is known for its speed in certain operations like bootstrapping. It is based on the Tiny Galois Switching Window (TGSW) [26, 27] scheme. Key points:

- Efficient for bootstrapping and some specific computations.
- Supports arbitrary computations on encrypted data.
- Active research library.
- Written in C++.

2.3.5 HEAAN

HEAAN (Homomorphic Encryption for Arithmetic of Approximate Numbers) [28] by the Cryptography Lab at Seoul National University. It implements an approximate homomorphic encryption scheme proposed by Cheon, Kim, Kim, and Song (CKKS) that supports arithmetic operations on encrypted real or complex numbers. HEAAN is written in C++ and necessitates a Linux OS. It has been used in our research and has the following key features:

- **Approximate Arithmetic Operations:** It enables approximate arithmetic operations like addition and multiplication on encrypted data without decryption, with controllable precision.
- **Complex/Real Number Plaintext Space:** Unlike other HE schemes that operate on integers or bits, HEAAN's plaintext space is the set of complex or real number vectors.
- **Encoding/Decoding Methods:** HEAAN employs efficient encoding and decoding

methods that exploit a ring isomorphism to represent complex/real vectors as polynomials.

- **Bootstrapping:** Later versions of HEAAN introduced a bootstrapping algorithm to enable arbitrary computation depths.

2.4 HEAAN functions

The HEAAN library works with three user-defined values - n, $logp$, $logq$. n determines the length of input data, when represented as an array. $logp$ and $logq$ correspond to precision bits and computational depth, respectively.

2.4.1 Encryption

Input data can be encrypted in two ways -

- `void encrypt(Ciphertext& cipher, std::complex<double>* vals, long n, long logp, long logq)`
- `void encrypt(Ciphertext& cipher, double* vals, long n, long logp, long logq)`

2.4.2 Decryption

to access results of computation, we can decrypt ciphertexts using -

```
std::complex<double>* decrypt(SecretKey& secretKey, Ciphertext& cipher)
```

2.4.3 Addition

The addition operation can be performed between two ciphertexts or a ciphertext and constant -

1. `void add(Ciphertext& res, Ciphertext& cipher1, Ciphertext& cipher2)`
2. `void addConst(Ciphertext& res, Ciphertext& cipher, double cnst, long logp)`
3. `void addConst(Ciphertext& res, Ciphertext& cipher, std::complex<double> cnst, long logp)`

2.4.4 Multiplication

Multiplication shares similarities with addition in its operational characteristics.

- `void mult(Ciphertext& res, Ciphertext& cipher1, Ciphertext& cipher2)`
- `void multByConst(Ciphertext& res, Ciphertext& cipher, double cnst, long logp)`
- `void multByConst(Ciphertext& res, Ciphertext& cipher, std::complex<double> cnst, long logp)`
- `void multByConstVec(Ciphertext& res, Ciphertext& cipher, std::complex<double>* cnstVec, long logp)`

2.4.5 Power of x

HEAAN uses Taylor series expansion to compute x^n, where n is represented by *degree* parameter.

`void power(Ciphertext& res, Ciphertext& cipher, long logp, long degree)`

2.4.6 Exponent

e^x is approximated using its Taylor series representation in HEAAN. The *degree* parameter corresponds to the the number of Taylor coefficients to be considered for the approximation, $degree = 10$ is recommended.

```
void function(Ciphertext& res, Ciphertext& cipher, "EXPONENT", long logp,
long degree)
```

Chapter3

Face Analytics

3.1 Prelude

PolyProtect [29], a template protection scheme, transforms face embeddings into more secure templates using multivariate polynomials with user-specific parameters. Our research demonstrates a symbiotic relationship between FHE and PolyProtect in ensuring optimal security measures. The synergy between these two techniques is essential, as each contributes unique strengths that, when combined, establish a robust security framework. According to the inversion attack analysis conducted by PolyProtect [29], the risk of reversibility is notably high (30 - 99%) when an attacker possesses multiple (more than 4) templates of the same face. Even with the compromise of a single template, there is a 95% likelihood of reversibility when an *overlap* of 4 or greater is employed. However, PolyProtect introduces a tradeoff between accuracy and security, wherein increasing the *overlap* parameter enhances accuracy in face recognition and analysis tasks but concurrently diminishes template security. To address this challenge, we implement a strategy of encrypting the face embeddings using FHE and then applying the PolyProtect template with high overlap. This proactive step ensures that even under the full disclosure threat model, irreversibility, unlinkability and prevention of soft biometrics leakage are ensured without compromising identification performance.

Conversely, the integration of PolyProtect into our approach serves as a countermeasure against threats targeting compromised FHE systems, such as secret key leaks and passive

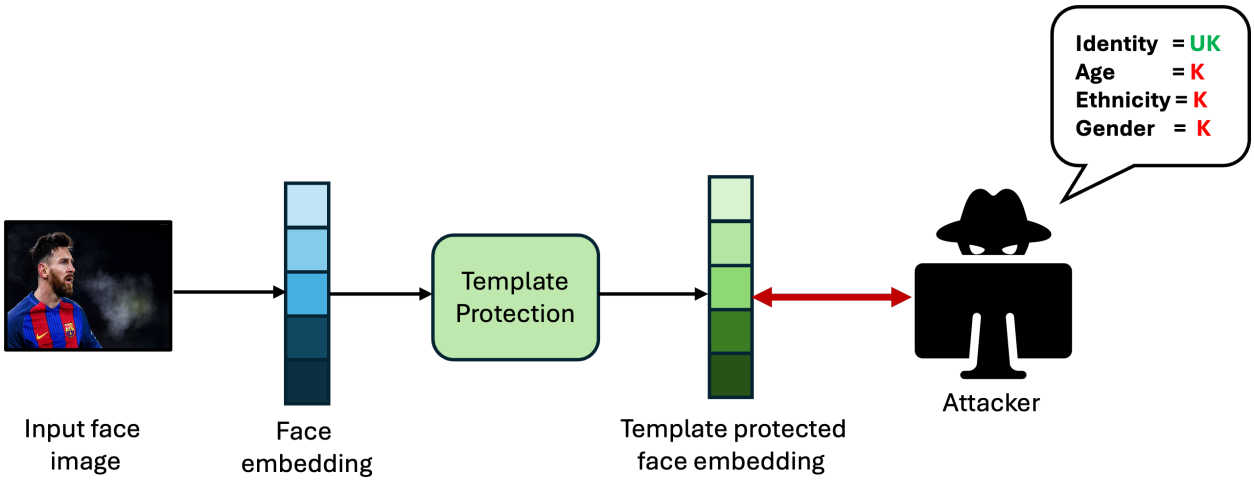


Figure 3.1: An overview of the implementation of the state-of-the-art method - Template Protection. Template Protection protects against embedding inversion attacks (identity) but soft-biometric features are exposed (K - Known; UK - Unknown).

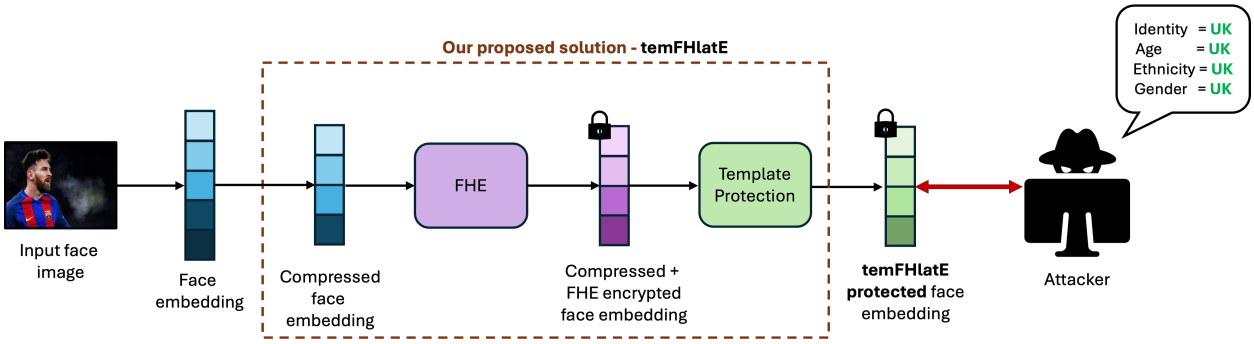


Figure 3.2: An overview of the implementation of our solution - temFHlatE. TemFHlatE protects against both embedding inversion attacks (identity) and leakage of soft-biometric features (UK - Unknown).

attacks outlined in [30]. The identified risks pertain to FHE systems engaged in machine learning computations, such as mean and variance calculations. By combining both FHE and PolyProtect techniques (Fig. 3.2), our approach offers a more comprehensive and resilient privacy solution compared to relying on either method in isolation.

3.2 Threat Model

We use the term *facial analytics* to refer to the process of deducing semantic information from a face image. This could include sensitive information such as age, gender, ethnicity, and health [31] – sometimes known as soft biometrics. The possibility of deducing soft biometric cues from face images or their embeddings using automated techniques is a source of concern. These automated techniques can be machine learning models such as SVMs or deep neural networks (DNNs). For example, a face image or its embedding can be "stolen" by hackers and various soft biometrics can be derived from them thereby revealing sensitive information.

In our work, we presume that the face embedding is provided in an encrypted form. Our goal is to ensure that the encrypted embedding does not reveal any soft biometric information to unauthorized users. Note that the threat remains unchanged even if the parameters of the models used for extracting soft biometric information (e.g., weights of a DNN) are encrypted.

We consider the most challenging threat model, as in Fig. 3.3, according to ISO/IEC 30316, which is the full disclosure model. The attacker possesses complete knowledge of the PolyProtect method [29], including its algorithm, number of embedding elements (m), user-specific parameters (e.g., C, *overlap*, and E), and one or more PolyProtected templates corresponding to a face embedding. In addition, we assume that the public key used in FHE is available but not the private key. If the embeddings are *not* encrypted, the hacker can infer soft biometric information from the PolyProtect template as we show in Table 3.3.

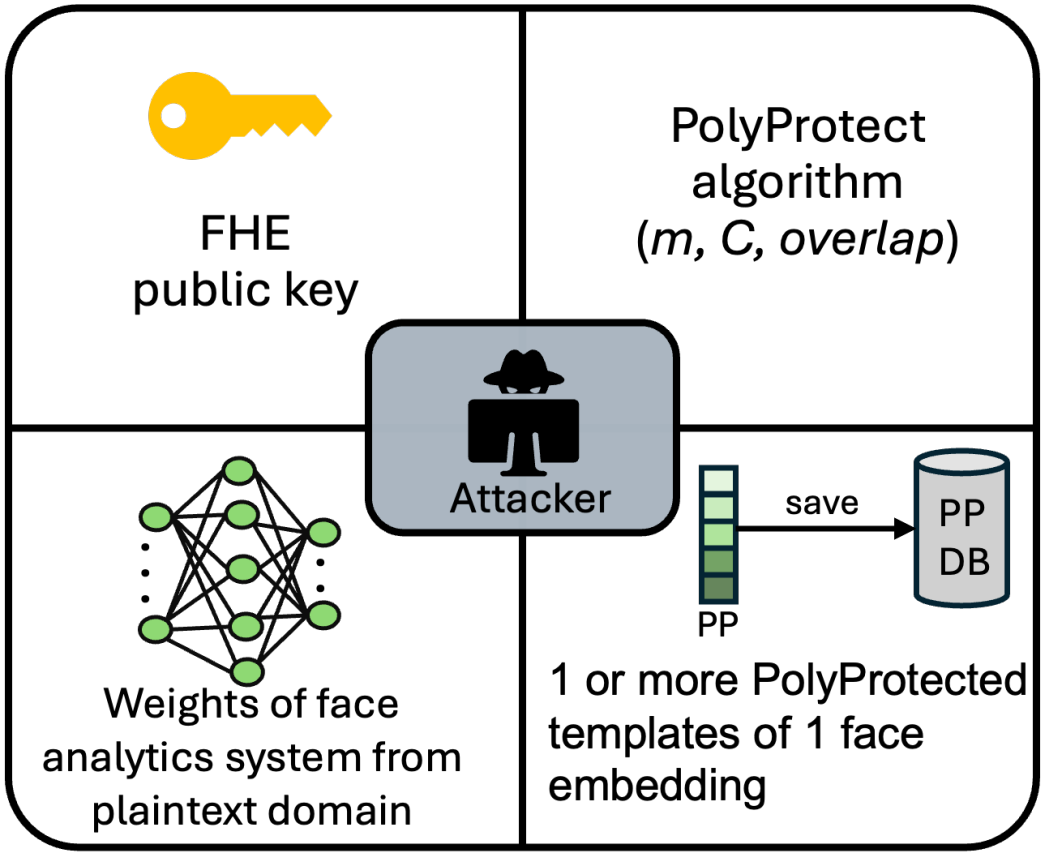


Figure 3.3: An overview of the threat model being considered in our research.

3.3 Related Work

A notable body of literature explores privacy enhancements to soft biometrics at both the image and embedding (template) levels in face recognition. PFRNet [32] uses an Autoencoder framework to disentangle identity from attribute information to suppress gender information in face embeddings. Similarly, SensitiveNets [33] uses a privacy-preserving neural network that suppresses soft biometrics attributes. The approach adopts an adversarial regularizer, which incorporates a sensitive information removal function into the learning objective. The Multi Incremental Variable Elimination (Multi-IVE) method [34] works by eliminating those feature variables in embeddings that predict soft biometric attributes. Increasing the number of eliminations was shown to decrease the soft biometrics leakage but significantly affected the identification performance. In [35], the authors introduce an adversarial attack approach

designed to protect gender information in facial images. The method involves perturbing the image to minimize the estimated mutual information between the feature distribution acquired from a face recognition network and the gender variable. This method reduces gender leakage (prediction accuracy) by an average of 91% to 80% across three datasets. Reversible Attribute Privacy Preservation (RAPP) [36] uses a stream cipher to determine the sensitive attributes that have to be concealed with a user-defined password; it supports recovering the original attributes. It also uses an attribute adversarial network to generate perturbed images that conceal various attributes while retaining the utility of face verification. However, the identification performance is negatively impacted. The authors work around this challenge by reducing the number of features being concealed and the intensity of concealment. PrivacyNet [37] is another technique to impart soft biometric privacy to face images while preserving recognition capabilities via image perturbation using a GAN-based Semi-Adversarial Network (SAN). PrivacyNet also allows a person to choose the specific facial attributes to be obfuscated while allowing the other attributes to be extracted. One of the drawbacks of this image perturbation technique is that it sometimes does not generate realistic images and cannot conceal soft biometric features from a human observer.

Although current approaches mitigate the leakage of soft biometric attributes, they do not suppress it to the level of a "random guess". In our work, we show that through the use of FHE, we can restrict this leakage in face embeddings to a level that is equal to or lower than that of a random guess. In [30], the authors show the susceptibility of privacy enhancement techniques such as homomorphic encryption. We address this susceptibility by employing a template protection scheme in addition to homomorphic encryption. Further, FHE encryption offers a stricter theoretical guarantee than existing methods for the security of soft biometrics.

FHE has been used in prior work for securing face recognition. Boddeti [38] proposed encrypting the face embeddings and performing face matching in the FHE domain. Batching and dimensionality reduction techniques are also explored to balance face-matching accu-

racy and computational complexity. In [39], the authors introduce an efficient approach for searching encrypted probe images against a large gallery, using fixed-length representations. In [40], the authors proposed a time-efficient and space-efficient face matching in the FHE domain for securing face templates. Our approach stands out from previous work by integrating a template protection scheme, a compression technique, and FHE to enhance the security of face templates. Moreover, we conducted thorough experiments to assess their efficacy in mitigating the leakage of soft biometrics.

3.4 Ablation Study

3.4.1 Variation of user-defined parameters (C, m, $overlap$)

We conducted experiments with various user parameters in PolyProtect to investigate their impact on the leakage of soft biometrics. As *overlap* increased from 0 to 3, there was a notable increase in age leakage, particularly for values $overlap > 0$. However, gender and ethnicity showed consistent levels of leakage across different values of *overlap* (Fig. 3.4(a)). As the overlap increases, the amount of embedding information retained after the PolyProtect transformation also increases, potentially resulting in a higher risk of soft biometric leakage. In PolyProtect, the parameter m dictates the number of terms in the polynomial. At $m = 6$, we observed maximum leakage for age, whereas for gender and ethnicity, maximum leakage occurred at $m = 7$. Conversely, selecting $m = 5$ minimized leakage across all three soft biometric attributes (Fig. 3.4(b)). Additionally, the parameter C, which defines the range of values for the polynomial coefficients [-C, C], was varied from 10 to 60 in our study. The leakage of ethnicity remained relatively stable across all tested values of C, while age exhibited an increase from $C = 10$ to 20, and gender showed a slight decrease from $C = 45$ to 60 (Fig. 3.4(c)).

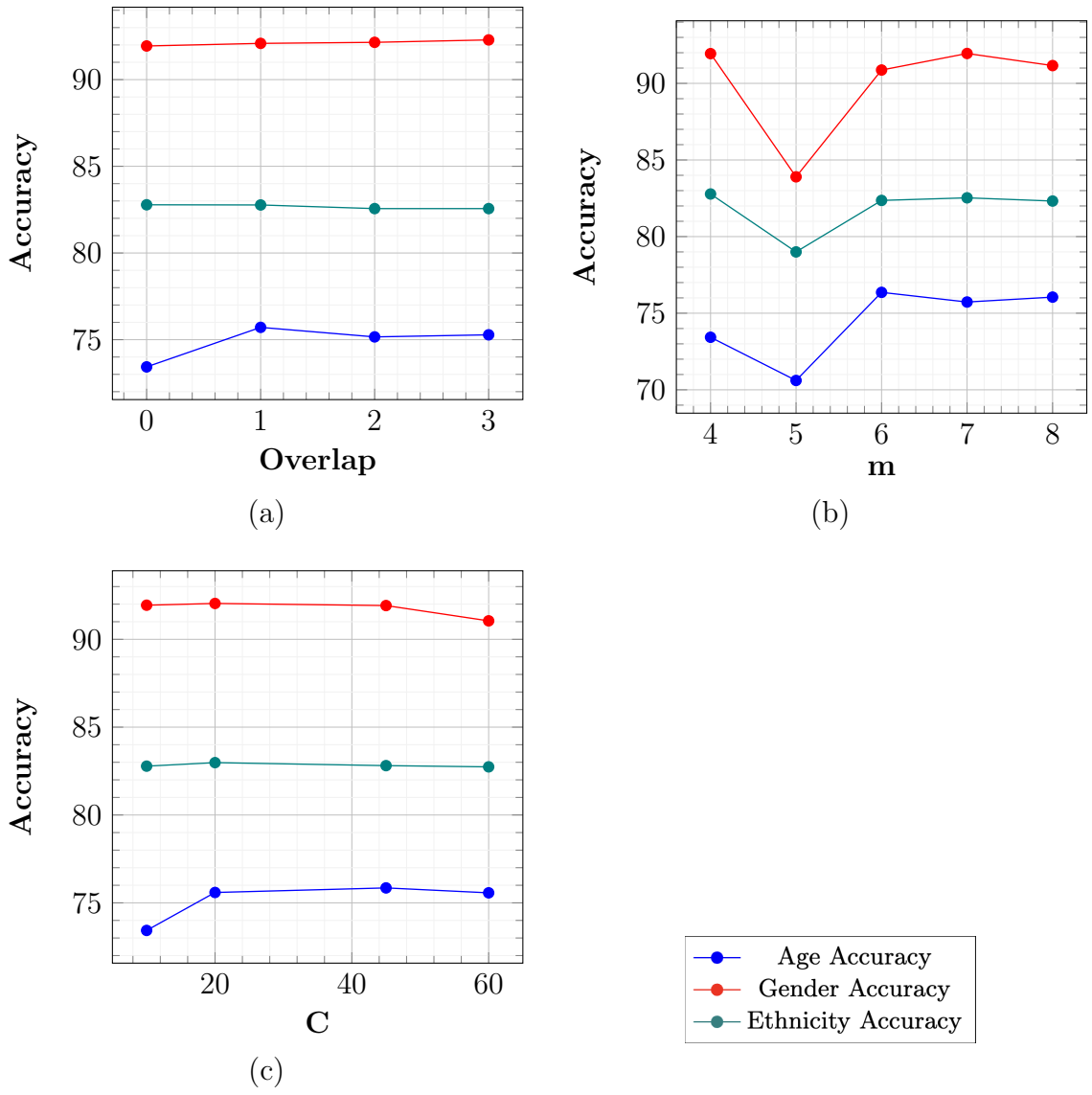


Figure 3.4: Ablation Study with the PolyProtect parameters shows soft biometrics leakage in different settings. (a) Overlap (b) m - Length of Polynomial Coefficients/Exponents (c) [-C,C] - Range of polynomial coefficients.

3.4.2 Summation of Ciphertext elements

Implementing PolyProtect, cosine similarity, and fully connected layers in FHE requires summing up the elements within a ciphertext. This is not straightforward as we cannot access individual elements of a ciphertext. We have performed a comparison between the three approaches based on the time taken with different input sizes as shown in Fig. 3.5. The three approaches to efficiently achieve the summation are described below:

3.4.2.1 Naive Rotation

This is a brute-force method for summation within a ciphertext where expensive ciphertext rotations are performed $N-1$ times and the running sum is computed until all the elements are covered (Algorithm 1).

3.4.2.2 Discrete Fourier Transform

When the Discrete Fourier Transform (DFT) of a signal is computed, the first value of DFT or the DC component will give the sum of the input signal values. We use this property to calculate the DFT of the ciphertext and get the sum of its values.

3.4.2.3 Fold and Add

This is a more efficient version of the naive rotation method described earlier, which can be visualized as iteratively folding the array into half and adding the corresponding folded parts $\log_2 N - 1$ times as described in algorithm 2 [41]. Fig. 3.5 shows that the *Fold and Add* method was the fastest among the three with a significant speedup than the naive rotation method, especially for large ciphertext sizes.

Algorithm 1 Add Ciphertext Elements Through Left Rotation $n-1$ times

0: **function** NAIVE_ADD(Ciphertext c, long N)
0: Ciphertext $c1 \leftarrow c$
0: **while** $N > 0$ **do**
0: $LeftRot(c1, 1)$ {Left Rotates Ciphertext by 1}
0: $c \leftarrow Add(c, c1)$ {Adds two Ciphertexts}
0: $N \leftarrow N - 1$
0: **end while**
0: **return** c
0: **end function**=0

Algorithm 2 Add Ciphertext Elements Through Left Rotation $\log_2 N$ times

0: **function** FOLD_ADD(Ciphertext c, long N)
0: $k \leftarrow \lceil \log_2 N \rceil$
0: $i \leftarrow k - 1$
0: **while** $i > 0$ **do**
0: Ciphertext $c1 \leftarrow LeftRot(c, 2^i)$
0: $c \leftarrow Add(c, c1)$
0: $i \leftarrow i - 1$
0: **end while**
0: **return** c
0: **end function**=0

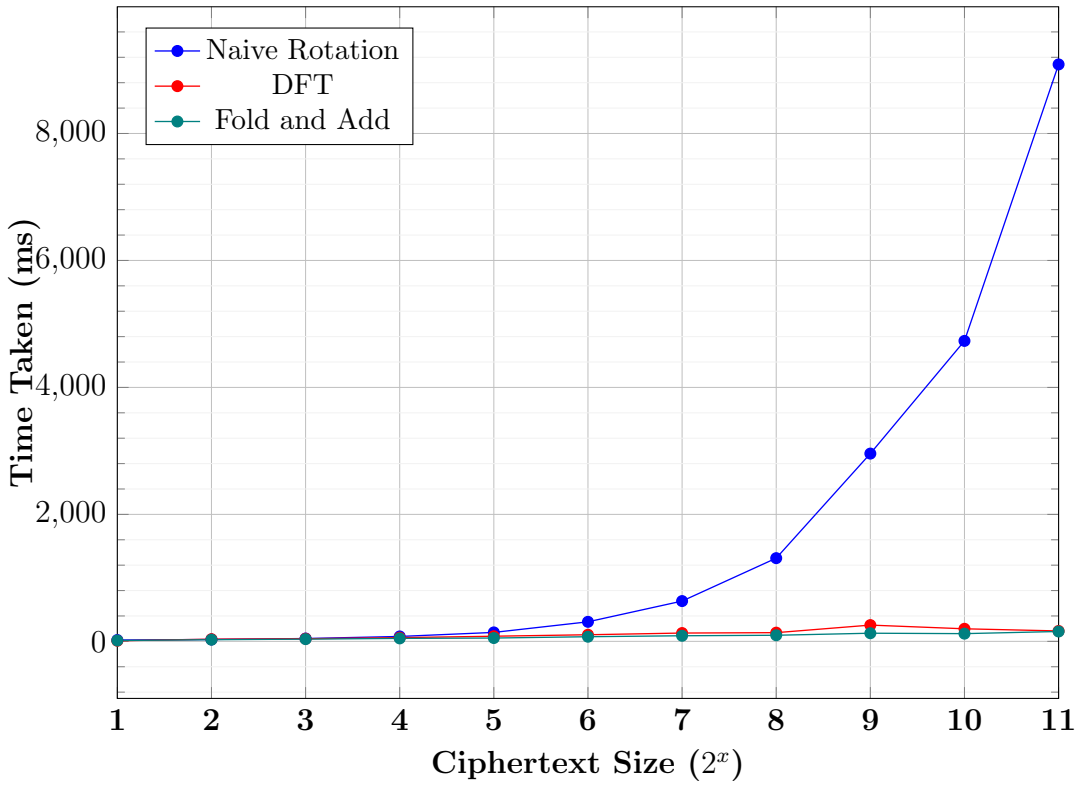


Figure 3.5: Execution time to compute summation of elements within a ciphertext.

3.5 Datasets

We have performed our experiments on CelebSet [42] and Balanced Faces in the Wild (BFW) [43]. The statistics of these datasets have been detailed in Tables 3.1, 3.2. As the BFW dataset lacks age annotations for its face images, we utilized a pre-trained model trained on the CelebSet dataset to predict the ages of the BFW images. These predicted ages were then employed in our experiments.

Table 3.1: Statistics of the CelebSet Dataset (80 Identities).

Gender		**Age**				**Ethnicity**			
Males	**Females**	**0-22**	**23-40**	**41-59**	**60+**	**Hispanic**	**White**	**Black**	**Asian**
38,080	34,409	5,279	43,357	22,781	1,072	738	57,873	13,414	464
52.50%	47.50%	7.28%	59.82%	31.43%	1.47%	1.01%	79.85%	18.50%	0.64%

3.6 Protection Techniques

Our primary objective is to elevate the privacy and security standards of face embeddings utilized in facial analytics, aiming to foster greater acceptance and trust within society. In our experimental setup, we assess the extent of soft-biometric information leakage within a face embedding across different scenarios. We perform our experiments on two face image datasets: (i) CelebSet [42], (ii) BFW [43], and use FaceNet [44] and AdaFace [45] to extract face embeddings. The datasets have been chosen such that they provide the ground truth for

Table 3.2: Statistics of the BFW Dataset (100 Identities; M - Males; F - Females; I - Indian; W - White; B - Black; A - Asian).

Gender		**Predicted Age**						**Ethnicity**			
M	**F**	**0-4**	**5-12**	**13-19**	**20-39**	**40-59**	**60+**	**I**	**W**	**B**	**A**
10,000	10,000	0	50	16,326	3,612	12	0	5,000	5,000	5,000	5,000
50%	50%	0%	0.25%	81.63%	18.06%	0.06%	0%	25%	25%	25%	25%

identity, and soft-biometric attributes (age, gender, and ethnicity). Soft-boimetric leakage is measured across three Template Protection techniques, namely, (i) PolyProtect; (ii) Negative Face Recognition; and (iii) Minimum Information Units.

3.6.1 Template Protection

Amongst the many existing protection templates, we have adopted **PolyProtect** [29] to showcase the benefits of our work. Let $V = [v_1, v_2, ..., v_n]$ denote an n-dimensional real-number face embedding. PolyProtect maps V to another real-numer feature vector, $P = [p_1, p_2, ..., p_k]$ (where $k < n$) as shown in Fig. 3.6. P is the PolyProtected template of V. m (where $m << n$) consecutive elements from V are mapped to single elements in P via a polynomial equation of coefficients, $C = [c_1, c_2, ..., c_m]$ and exponents, $E = [e_1, e_2, ..., e_m]$. C and E are user-defined, non-zero, and distinct for each user of the face recognition system. The first m elements of V are mapped to p_1 as :

$$p_1 = c_1 v_1^{e_1} + c_2 v_2^{e_2} + ... + c_m v_m^{e_m} \tag{3.1}$$

Another important user-defined parameter is *overlap*, which defines the number of common elements from V used in successive values in P. When $overlap = 0$, the elements of V in each set are unique. The minimum and maximum values for *overlap* are 0 and $m - 1$, respectively. The mapping for p_2 for overlaps 0 and $m - 1$, respectively, are as follows:

$$p_2 = c_1 v_{m+1}^{e_1} + c_2 v_{m+2}^{e_2} + ... + c_m v_{m+m}^{e_m} \tag{3.2}$$

$$p_2 = c_1 v_2^{e_1} + c_2 v_3^{e_2} + ... + c_m v_{m+1}^{e_m} \tag{3.3}$$

The authors of PolyProtect [29] have also performed an extensive survey on a number of existing Biometric Template Protection (BTP) methodologies and evaluated them based on recognition accuracy, irreversibility, and unlinkability [46]. According to the survey, none of

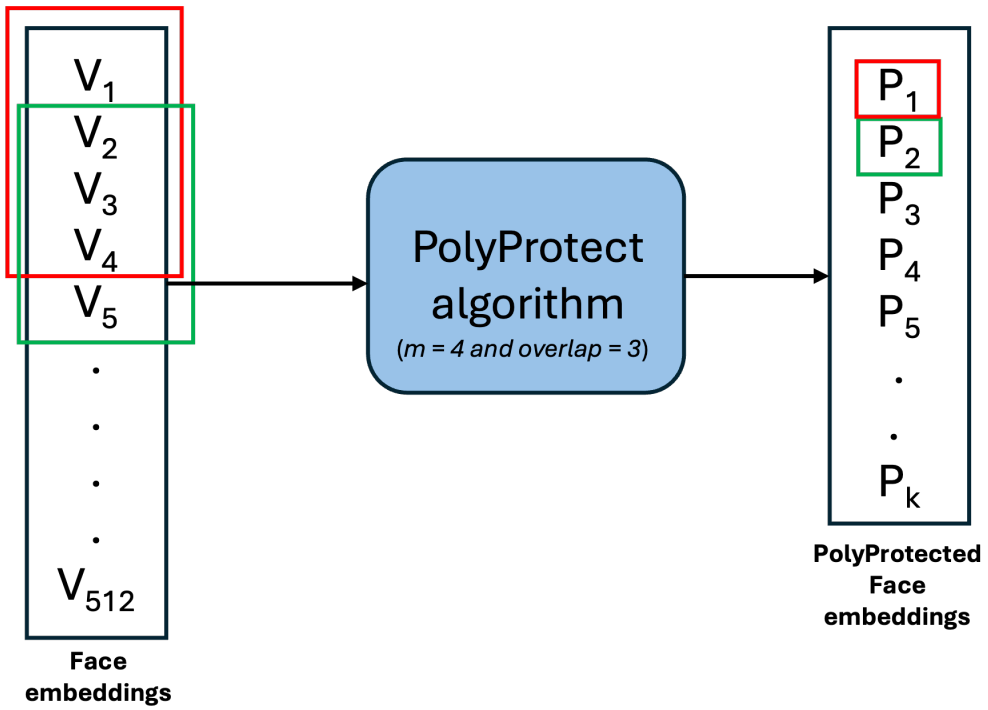


Figure 3.6: An overview of the working of the PolyProtect algorithm. For demonstration purposes, we have considered $m = 4$ and $overlap = 3$ in the figure.

the existing BTP methodologies before PolyProtect [29] satisfy all three criteria - recognition accuracy, irreversibility, and unlinkability.

3.6.2 Embedding Compression

It is commonly believed that compressing embeddings can improve privacy leakage. We explore embedding compression through **Matryoshka Representation Learning (MRL)** [47] as a technique to improve privacy, in addition to template protection and encryption. MRL is an innovative approach that enhances representation learning by encoding information at various granularities within a single embedding. We have used MRL in our work to compress face embeddings, extracted through FaceNet/AdaFace, from 512-dimension space to 64-dimension space. Fig. 3.7 shows that embeddings retain significant information until they are compressed to 64-dimension using MRL.

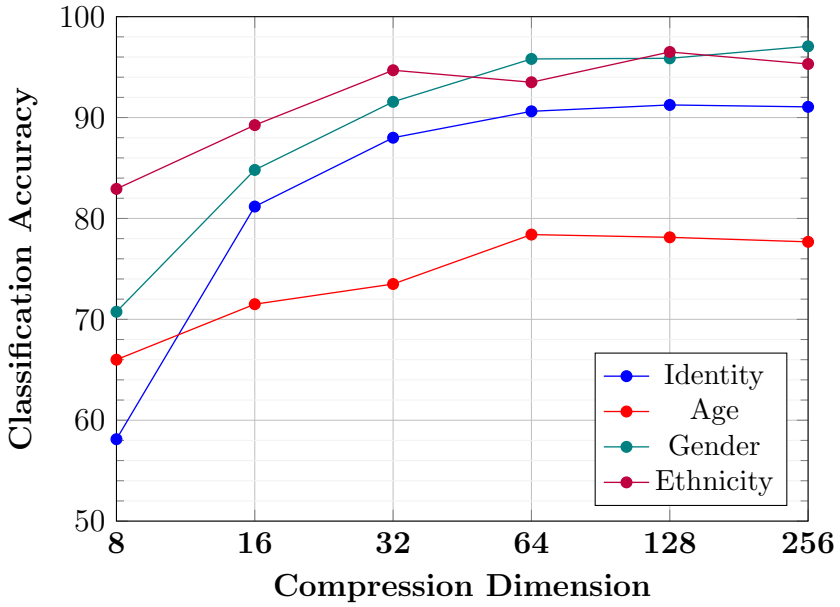


Figure 3.7: Performance of Matryoksha Representation Learning(MRL) in extracting features - Identity, Age, Gender, Ethnicity - from different compression dimensions. The graph is based on an experiment performed using AdaFace with CelebSet dataset.

3.6.3 TemFHlatE

Amongst the many existing protection templates, we have adopted **PolyProtect** [29] to showcase the benefits of our work. Let $V = [v_1, v_2, ..., v_n]$ denote an n-dimensional real-number face embedding. PolyProtect maps V to another real-numer feature vector, $P = [p_1, p_2, ..., p_k]$ (where $k < n$). P is the PolyProtected template of V. m (where $m << n$) consecutive elements from V are mapped to single elements in P via a polynomial equation of coefficients, $C = [c_1, c_2, ..., c_m]$ and exponents, $E = [e_1, e_2, ..., e_m]$. C and E are user-defined, non-zero, and distinct for each user of the face recognition system. The first m elements of V are mapped to p_1 as :

$$p_1 = c_1 v_1^{e_1} + c_2 v_2^{e_2} + ... + c_m v_m^{e_m} \tag{3.4}$$

Another important user-defined parameter is *overlap*, which defines the number of common elements from V used in successive values in P. When $overlap = 0$, the elements of V

in each set are unique. The minimum and maximum values for *overlap* are 0 and $m-1$, respectively. The mapping for p_2 for overlaps 0 and $m-1$, respectively, are as follows:

$$p_2 = c_1 v_{m+1}^{e_1} + c_2 v_{m+2}^{e_2} + ... + c_m v_{m+m}^{e_m} \tag{3.5}$$

$$p_2 = c_1 v_{2}^{e_1} + c_2 v_{3}^{e_2} + ... + c_m v_{m+1}^{e_m} \tag{3.6}$$

The authors of PolyProtect [29] have also performed an extensive survey on a number of existing Biometric Template Protection (BTP) methodologies and evaluated them based on recognition accuracy, irreversibility, and unlinkability [46]. According to the survey, none of the existing BTP methodologies before PolyProtect [29] satisfy all three criteria - recognition accuracy, irreversibility, and unlinkability.

We propose a combination of FHE and PolyProtect, called **temFHlatE**, to achieve soft-biometric privacy and preserve identification performance. The face embeddings are first encrypted as a single ciphertext. Parts of the ciphertext (specifically m elements) undergo a multivariate polynomial transformations to form a single element of the protected embedding.

Performing non-linear operations in the encrypted domain is not as straightforward as in the plaintext domain. Numerous adaptations are necessary in the encrypted domain as shown in Fig. 3.9. Additionally, FHE ciphertexts do not allow single element access. Detailed overview of the state-of-the-art pipeline in comparison to our pipeline is shown in Fig. 3.1 and Fig. 3.2 respectively.

3.7 Experiments

In the plaintext domain, we take a supervised learning approach to learn face features from face embeddings and classify them according to the provided labels. We use a simple neural network with 4 different classification heads for identity, age, gender and ethnicity as shown in Fig. 3.8.

On the other hand, as of today, FHE ciphertexts cannot be trained using neural networks.

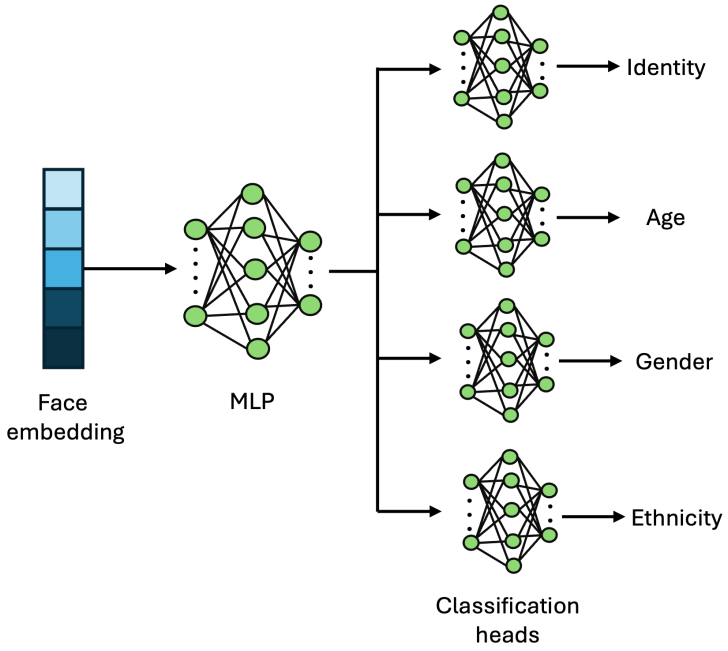


Figure 3.8: Face identification and soft-biometric prediction in the plaintext domain using 4 separate classification heads.

FHE allows us to perform face analytics using the ciphertexts in the encrypted domain, but, significant adaptations are necessary to implement non-linear functions.

3.7.1 Face identification

As in Fig. 3.9, to perform face identification, we conduct a 1:N search in the database of temFHlatE protected embeddings using cosine distance between the embeddings as a metric. Cosine distance between two embeddings can be defined as -

$$\text{Cosine Distance}(embedding_1, embedding_2) = \frac{embedding_1 . embedding_2}{|embedding_1|\,|embedding_2|} \tag{3.7}$$

Absolute value of an embedding, $|embedding|$ is necessary to compute cosine distance. The division operator and square-root function is not permissible in the encrypted domain. To circumvent this, we generate a polynomial approximation of the inverse square-root function using Polynomial Regression in the unencrypted domain. We restrict the input to the

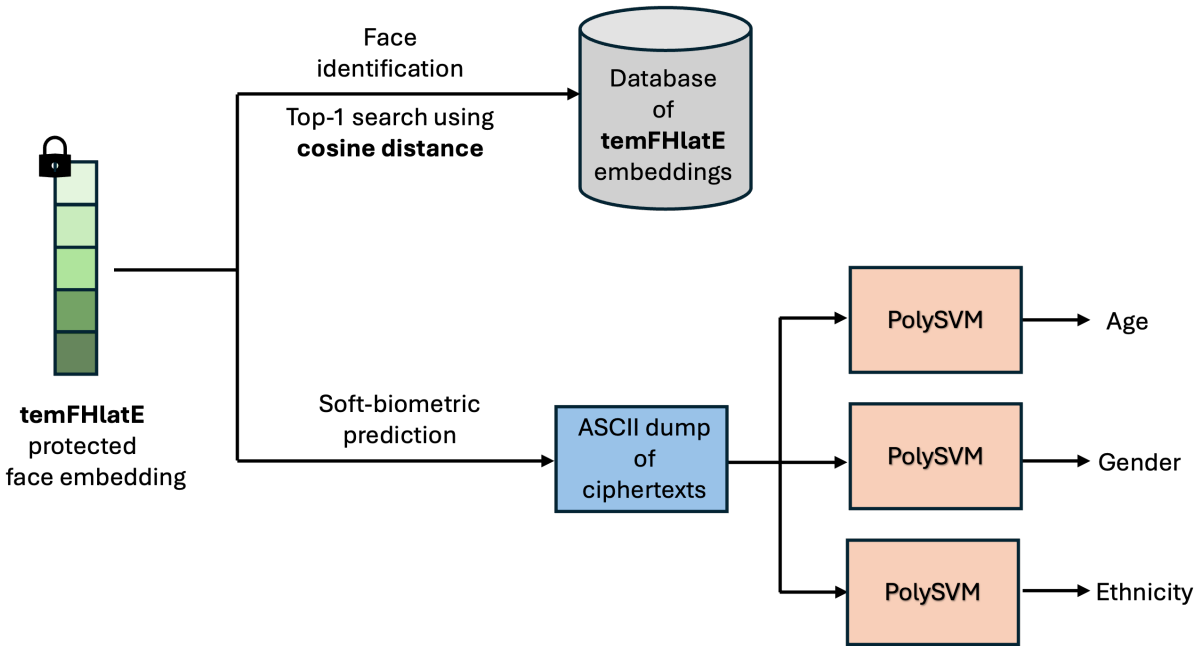


Figure 3.9: An overview of the necessary adaptations in the encrypted domain to perform face analytics.

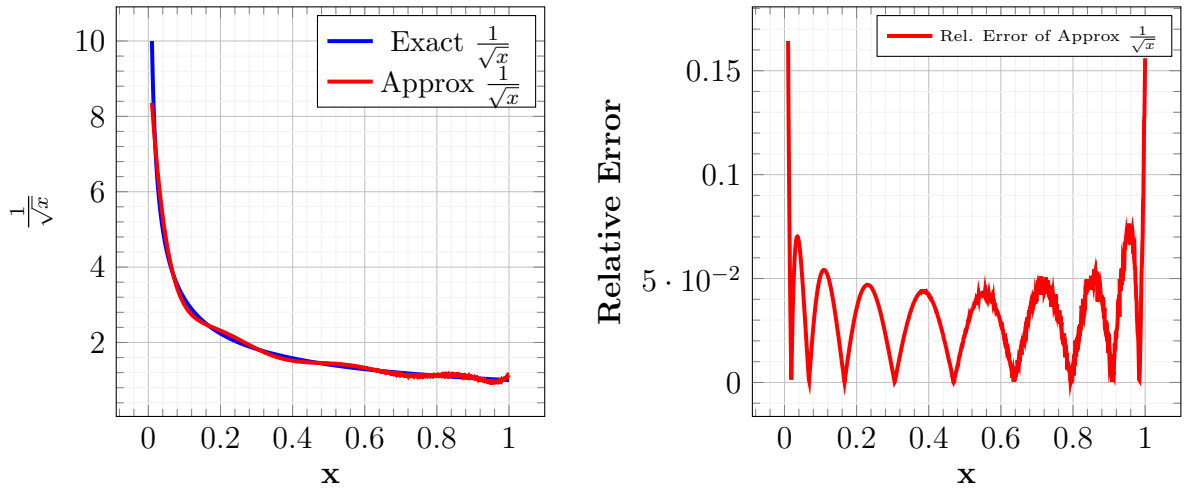
range (0, 1] to achieve a closer approximation. The performance of this approximation is measured through the relative error of 2000 random points in the range (0, 1]. We consider both 6-degree and 8-degree polynomials as a tradeoff between computational depth and accuracy, as in Fig. 3.10.

3.7.2 Soft-biometric prediction

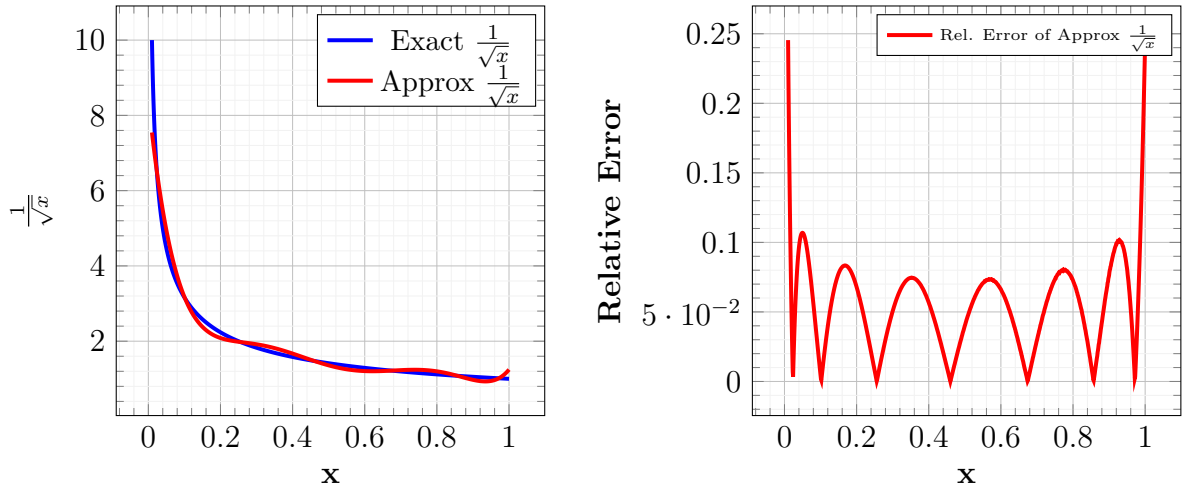
On the other hand, for soft-biometric prediction, we extract the ASCII dump of the ciphertexts and use polySVMs to predict their labels.

3.8 Results

To evaluate our solution against the existing solutions, we use two metrics that measure the gain in privacy, where R_o and R_p represent the recognition performances on the original data



(a)



(b)

Figure 3.10: (a) 6-degree polynomial (b) 8-degree polynomial approximation of inverse square root and its relative error over 2000 random points in the range (0,1].

and the privacy-enhanced data, respectively.

$$\text{Privacy Gain (PG)} = (1 - R_p) - (1 - R_o) \tag{3.8}$$

$$\text{Suppression Rate (SR)} = \frac{R_o - R_p}{R_o} \tag{3.9}$$

A positive value of Privacy Gain signifies enhanced data protection. Whereas, in case of Suppression Rate, a higher value indicates higher privacy.

As evident from Tables 3.3, 3.4, 3.5 our proposed approach prevents the leakage of soft biometrics from face embeddings with minimal loss in identification accuracy (<2.5%), high Privacy Gain and high Suppression Rate. We can observe that our approach reduced the classification accuracies of soft biometric attributes to the level of random chance across the two datasets. We could also achieve an almost ideal Privacy Gain and Suppression Rate in certain scenarios, but we believe this could be because of the imbalanced nature of our datasets. Our experiments showcase that a combination of MRL, FHE, and PolyProtect in this order yields maximum protection against soft biometric leakage (Fig. 3.11 and Fig. 3.12).

The consistent efficacy of our method across two different embeddings (FaceNet and AdaFace) and datasets (CelebSet and BFW) shows the ability of our method to generalize in different conditions. We also prove the ability of FHE to work seamlessly with template protection schemes and embedding compression techniques for preserving the privacy of face templates.

Additionally, we find that utilizing a soft biometrics classifier network trained on plaintext data allows for seamless inference within the FHE domain, yielding the same predictive performance as that in the unencrypted domain (as depicted in Table 3.3 in "None"). This underscores the feasibility of conducting not only identification but also soft biometric analysis in the secure FHE domain.

Table 3.3: Face identification and soft biometric classification accuracy (MRL - Matryoksha Representation Learning; FHE - Fully Homomorphic Encryption). Note that the proposed approach retains identification accuracy while successfully reducing soft biometric classification accuracy.

Embeddings	**Dataset**	**Template Protection**	**Id**	**Gender**	**Age**	**Ethn**
FaceNet	CelebSet	None	99.42%	98.12%	87.68%	98.81%
		PolyProtect	99.42%	97.35%	85.00%	98.06%
		MRL	97.93%	98.00%	85.87%	97.38%
		MRL + PolyProtect	97.00%	96.32%	87.32%	98.44%
		MRL + FHE	97.83%	**52.22%**	**6.12%**	**8.01%**
		TemFHlatE	96.95%	**52.22%**	**6.12%**	8.03%
	BFW	None	85.06%	95.25%	94.40%	91.97%
		PolyProtect	85.04%	95.07%	93.97%	91.62%
		MRL	84.42%	92.02%	93.98%	91.65%
		MRL + PolyProtect	84.31%	90.00%	87.90%	87.70%
		MRL + FHE	84.38%	49.98%	28.00%	24.01%
		TemFHlatE	84.28%	**49.50%**	**27.82%**	**24.00%**
AdaFace	CelebSet	None	99.41%	96.93%	90.25%	96.31%
		PolyProtect	99.38%	95.75%	90.18%	96.56%
		MRL	97.95%	97.63%	85.50%	98.94%
		MRL + PolyProtect	97.59%	96.82%	89.13%	98.25%
		MRL + FHE	97.91%	**52.22%**	6.12%	8.02%
		TemFHlatE	97.55%	**52.22%**	**6.11%**	**8.01%**
	BFW	None	88.55%	89.97%	85.02%	81.72%
		PolyProtect	88.46%	84.07%	84.62%	74.62%
		MRL	88.33%	87.55%	83.83%	76.43%
		MRL + PolyProtect	88.29%	85.58%	83.66%	75.58%
		MRL + FHE	88.23%	49.98%	27.97%	24.02%
		TemFHlatE	88.21%	**49.50%**	**27.90%**	**24.00%**

Table 3.4: Privacy Gain across different soft biometric attributes.

Embeddings	**Dataset**	**Template Protection**	**Id**	**Gender**	**Age**	**Ethn**
FaceNet	CelebSet	PolyProtect	99.42%	0.72	2.68	0.75
		MRL	97.93%	0.12	1.81	1.43
		MRL + PolyProtect	97.00%	1.80	0.36	0.37
		MRL + FHE	97.83%	**45.90**	**81.56**	**90.80**
		TemFHlatE	96.95%	**45.90**	**81.56**	90.78
	BFW	PolyProtect	85.04%	0.18	0.43	0.35
		MRL	84.42%	3.23	0.42	0.32
		MRL + PolyProtect	84.31%	5.25	6.50	4.27
		MRL + FHE	84.38%	45.27	66.40	67.96
		TemFHlatE	84.28%	**45.75**	**66.58**	**67.97**
AdaFace	CelebSet	PolyProtect	99.38%	1.18	0.07	-0.25
		MRL	97.95%	-0.7	4.75	-2.63
		MRL + PolyProtect	97.59%	0.11	1.12	-1.94
		MRL + FHE	97.91%	**44.71**	84.13	88.29
		TemFHlatE	97.55%	**44.71**	**84.14**	**88.30**
	BFW	PolyProtect	88.46%	5.90	0.40	7.10
		MRL	88.33%	2.42	1.22	5.29
		MRL + PolyProtect	88.29%	4.39	1.36	6.14
		MRL + FHE	88.23%	39.99	57.05	57.70
		TemFHlatE	88.21%	**40.47**	**57.12**	**57.72**

Table 3.5: Suppression Rate across different soft biometric attributes.

Embeddings	**Dataset**	**Template Protection**	**Id**	**Gender**	**Age**	**Ethn**
FaceNet	CelebSet	PolyProtect	99.42%	0.0073	0.0306	0.0075
		MRL	97.93%	0.0012	0.0206	0.0145
		MRL + PolyProtect	97.00%	0.0183	0.0041	0.0037
		MRL + FHE	97.83%	**0.4678**	**0.9302**	**0.9189**
		TemFHlatE	96.95%	**0.4678**	**0.9302**	0.9187
	BFW	PolyProtect	85.04%	0.0019	0.0046	0.0038
		MRL	84.42%	0.0339	0.0044	0.0035
		MRL + PolyProtect	84.31%	0.0551	0.0689	0.0464
		MRL + FHE	84.38%	0.4753	0.7034	0.7389
		TemFHlatE	84.28%	**0.4803**	**0.7053**	**0.7390**
AdaFace	CelebSet	PolyProtect	99.38%	0.0122	0.0008	-0.0026
		MRL	97.95%	-0.0072	0.0526	-0.0273
		MRL + PolyProtect	97.59%	0.0011	0.0124	-0.0201
		MRL + FHE	97.91%	**0.4613**	0.9322	0.9167
		TemFHlatE	97.55%	**0.4613**	**0.9323**	**0.9168**
	BFW	PolyProtect	88.46%	0.0656	0.40	0.0869
		MRL	88.33%	0.0269	0.0143	0.0647
		MRL + PolyProtect	88.29%	0.0488	0.0159	0.0751
		MRL + FHE	88.23%	0.4444	0.6710	0.7061
		TemFHlatE	88.21%	**0.4498**	**0.6718**	**0.7063**

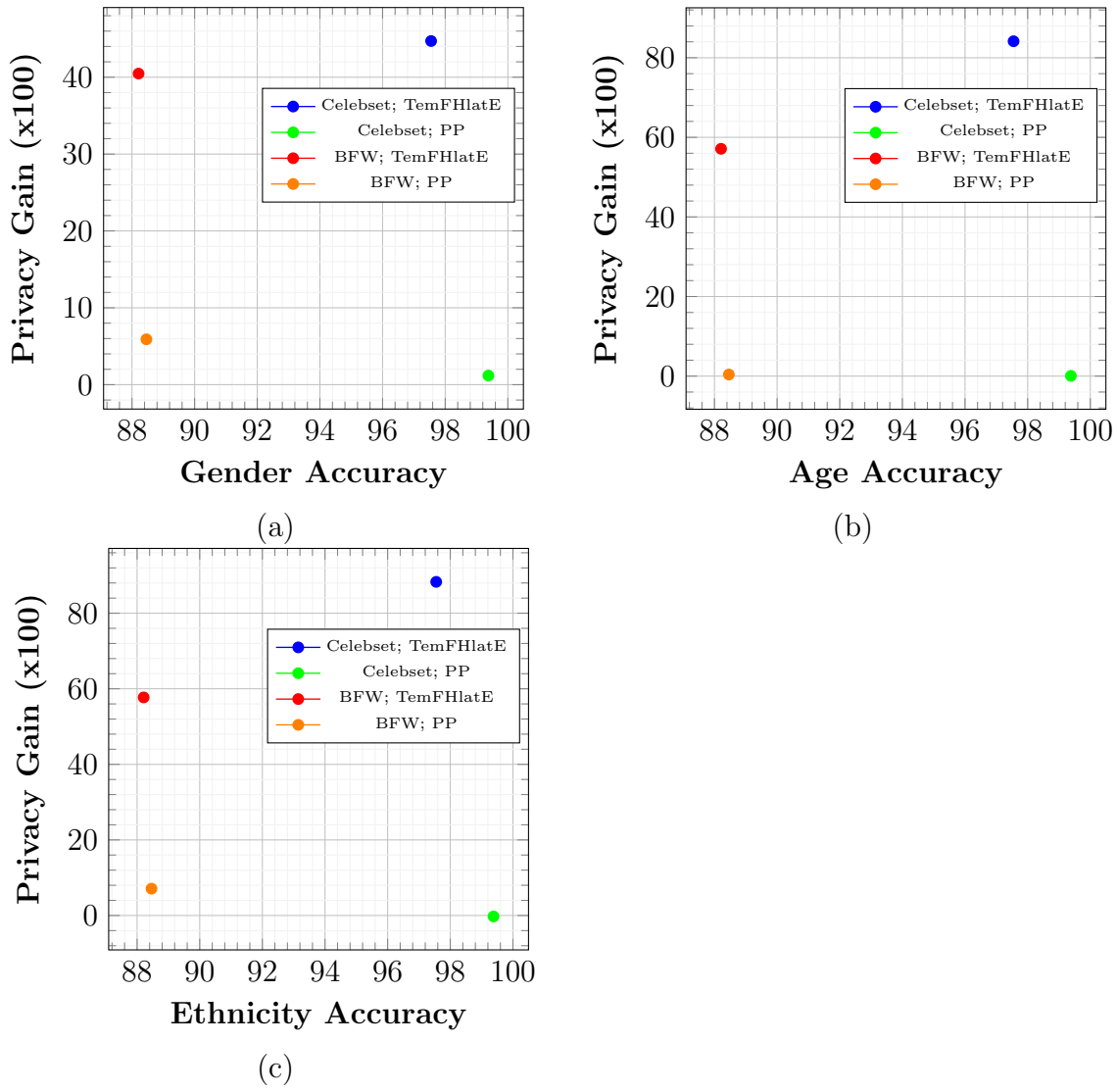


Figure 3.11: Privacy Gain of our proposed approach compared to baseline (PP) across different attributes - (a) Gender, (b) Age and (c) Ethnicity - using **Adaface** (MRL - Matryoksha Representation Learning; FHE - Fully Homomorphic Encryption; PP - PolyProtect).

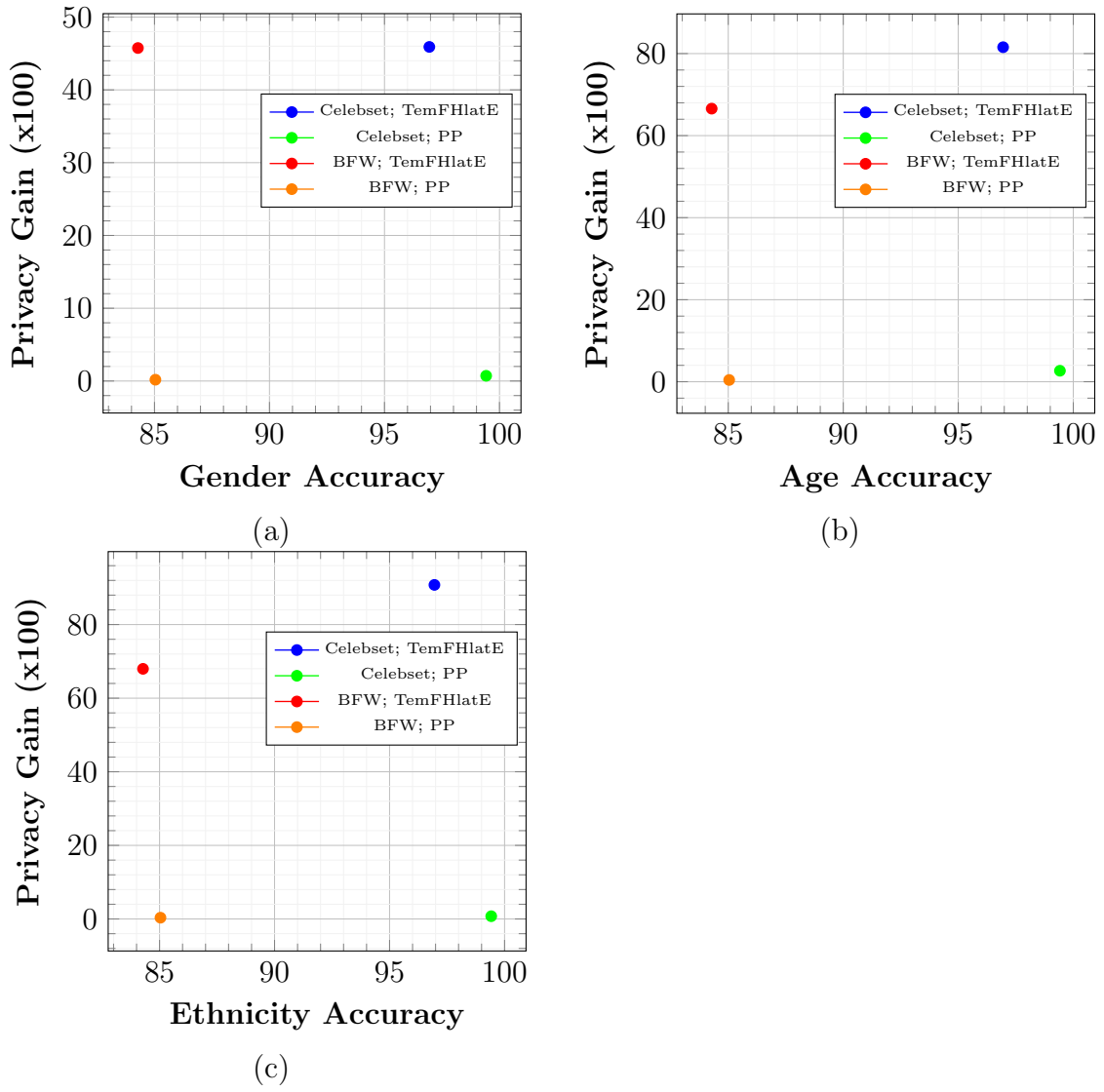


Figure 3.12: Privacy Gain of our proposed approach compared to baseline (PP) across different attributes - (a) Gender, (b) Age and (c) Ethnicity - using **FaceNet** (MRL - Matryoksha Representation Learning; FHE - Fully Homomorphic Encryption; PP - PolyProtect).

Chapter4

Autonomous UAVs

4.1 Prelude

Autonomous drones are exposed to various adversarial threats, such as - eavesdropping, traffic analysis, man-in-the-middle, and backdoor access [48]. From a deep learning perspective, attacks can be broadly classified into our types: membership inference, reconstruction, property inference, and model extraction [49]. In our research, we specifically address the scenario where an attacker can intercept communication between the drone and its navigation server, posing a potential risk to the UAV's secure operation. Our primary focus is on establishing secure and private communication channels for autonomous drone navigation.

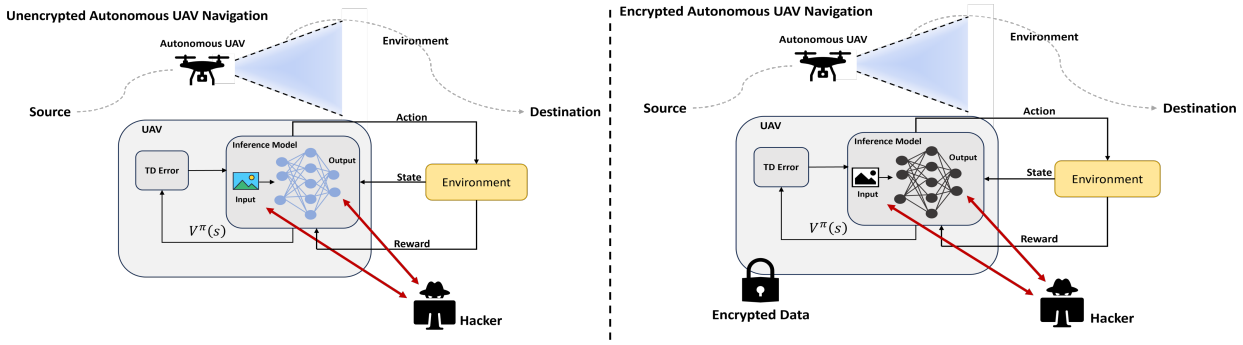


Figure 4.1: **Overview:** In an ordinary scenario the UAV is vulnerable to snooping attacks, as the attacker can directly steal the information. Or, query the model to infer target information, launching a model inversion attack. In our approach, the input is encrypted and the inference happens *in the encrypted domain*. Hence, the attacker is unable to exploit any meaningful information from the system. The figure has been adopted from [18].

When implemented properly, FHE helps achieve a high level of security. However, its implementation of mathematical operations is limited. To this end, in this work, we adapt an

RL model to the encrypted domain to facilitate the processing of encrypted real-time images captured by UAV cameras (Fig. 4.1). The RL model is adapted from [50] and utilizes the Actor-Critic policy within the Proximal Policy Optimization (PPO) algorithm. Key aspects of our approach include model compression using Knowledge Distillation, transforming convolutional layers into spectral domain operations, utilizing generalized matrix multiplication in fully connected layers, and customizing activation functions as polynomial approximations/comparators. Since the RL framework utilizes OpenAI Gym Library to derive the navigational steps from the extracted image features, we adapt the Library to the encrypted domain as well. A simple multi-layer perceptron is trained to replicate the OpenAI Gym library and its weights are used during inferencing in the encrypted domain. Remarkably, our end-to-end secure framework shows a negligible loss in performance.

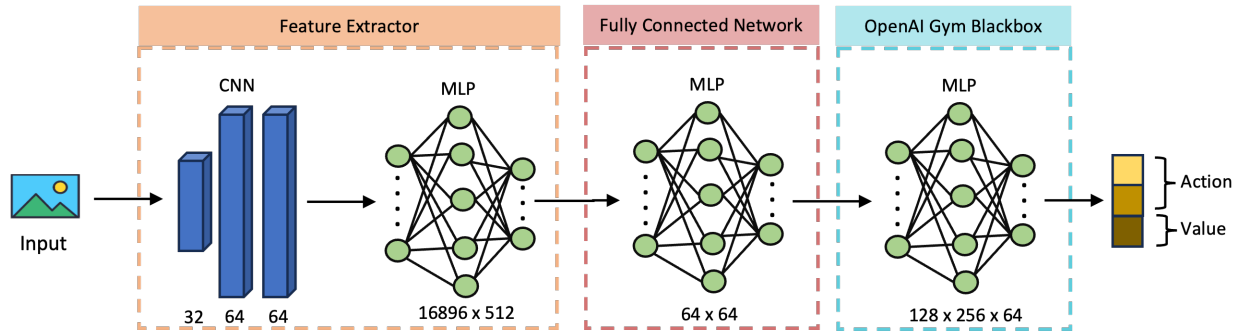


Figure 4.2: Architecture of the original model (Teacher Network).

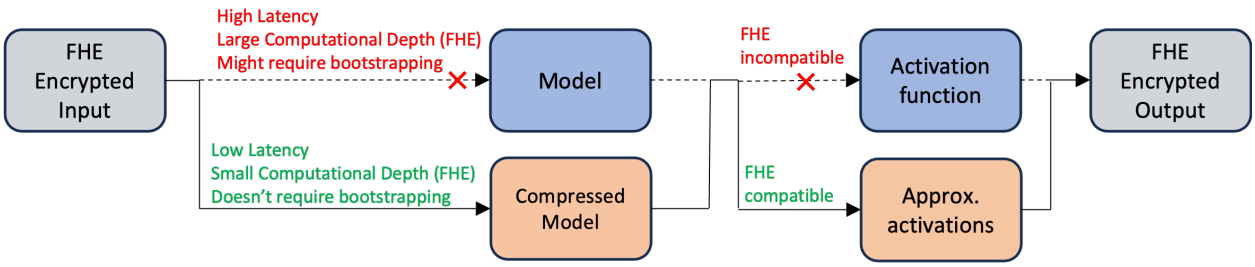


Figure 4.3: An overview of the need for an FHE optimized model.

4.2 Threat Model

Unmanned Aerial Vehicles (UAVs) deployed in critical scenarios are exposed to various adversarial threats, including (i) Data Poisoning, (ii) Model Inversion, and (iii) White-box attacks. In our research, we specifically address the scenario where an attacker can intercept communication between the drone and its navigation server, posing a potential risk to the UAV's secure operation. Our primary focus is on establishing secure communication channels between the drone and its navigation server, thereby safeguarding it against Targeted Attacks.

Our solution not only mitigates the risk of Targeted Attacks but also protects against Model Inversion attacks. This is achieved by the intelligent adaptation of different components of the model architecture to the encrypted domain. The server can be assumed to hold the weights of the model as matrices, and activation functions as polynomial approximations, instead of the true model architecture in sequence. Consequently, even with full knowledge of such weights, an attacker would be unable to configure the architecture, enhancing the security posture of the UAV system. Moreover, the overall execution of the algorithm happens on encrypted data. Thus one with access to the secret key can only consume the results. However, adversarial image attacks are not protected by this approach.

4.3 Related Work

Numerous surveys have delved into the privacy and security challenges specific to UAVs. Works such as [51] and [52] highlight the vulnerability landscape in UAV communication networks, emphasizing the delicate trade-off between robust security and the imperative for lightweight, efficient operations. These discussions underscore the crucial role of encryption in fortifying UAV systems against multifaceted threats, as presented by the authors in [53]. Our research aims to build upon these foundational insights, contributing to the ongoing

discourse on UAV security.

Homomorphic encryption has been employed in prior work to secure computations in the context of UAV navigation. For instance, in [54], the authors propose an extra key generation encryption technique using the Paillier Cryptosystem to prevent cipher data from being compromised. Further, Cheon et al. [55] explores the development of secure UAVs using a homomorphic public-key encryption method, enabling both secret communication and confidential computation. Another approach focuses on providing a secure and efficient method for third-party UAV controllers to collect and process client data, as demonstrated in [56]. The authors propose a Secure Homomorphic Encryption(SHE) framework, which transfers the FHE encryption to UAVs through an encryption protocol.

Despite notable progress in advancing autonomous systems and encryption methodologies for various applications [57, 58, 18], achieving a comprehensive and practical solution for secure drone systems has proven elusive. While previous works, such as [58], offer feasible frameworks for drone controllers, they do not address security at the drone level, leaving them vulnerable to attacks. Similarly, [18] presents a secure Reinforcement Learning-based framework for drone navigation, yet its practical implementation remains unfeasible. In contrast to the innovative approach of AutoFHE [59] for accelerating inference in encrypted domain of large CNN models (with a focus on ReLU amongst other activations), our work uses a small model with minimal activation functions.

Among various model compression techniques, including Pruning, Quantization, Decomposition, and Knowledge Distillation [60], our research finds Knowledge Distillation to be particularly effective for Fully Homomorphic Encryption (FHE). Pruning involves eliminating network components to create sparse models, which, although useful for acceleration and compression, doesn't significantly reduce computational time for CNNs in FHE. While Quantization typically operates in the BGV scheme, our research focuses on the CKKS scheme [21]. Although Decomposition shows promise, it doesn't match the effectiveness of reducing network depth through Knowledge Distillation.

4.4 Proposed Methodology

We have adopted a Reinforcement Learning-based model from [50] to showcase the effectiveness of FHE in providing private and secure autonomous navigation in UAVs. The model utilizes the Actor-Critic policy within the Proximal Policy Optimization (PPO) algorithm, capable of seamlessly operating on real-time video feeds captured by UAV cameras. In this report, we adapt this RL framework to operate in the encrypted domain.

The drone is trained using the Actor-Critic Reinforcement Learning algorithm [50] [61]. During training, both the Actor and Critic networks are utilized, whereas, during inferencing, only the Actor network is leveraged. The network architecture can be divided into two segments - Feature Extractor and Fully Connected Network as shown in Figure 4.1. The Feature extractor consists of three convolution blocks and one linear block as shown in Figure 4.2. Each convolution block consists of a Convolution layer, Batch Normalization layer, and ReLU activation layer. The linear block consists of a Dense Layer, Batch Normalization layer, and ReLU activation layer. The Fully Connected Network segment consists of two shared linear blocks (shared between Actor and Critic) and an output linear block as in Figure 4.2. The shared linear blocks are made up of a dense layer and utilize the TanH activation function.

Computation within the Fully Homomorphic Encryption (FHE) domain introduces several significant limitations as in Fig. 4.3, including the absence of individual element access in encrypted arrays, restricted computation depth, heightened time complexity, and the absence of inherent support for non-linear functions. Consequently, we choose to train the Actor-Critic model in the unencrypted domain with data generated in a simulated environment, employing Microsoft's AirSim library and Unreal Engine. Subsequently, leverage the model weights for inference within the encrypted domain. To achieve this, we carefully adapt each component of the Actor-Critic network to seamlessly operate within the FHE domain, addressing specific challenges presented by FHE. In addition to computational constraints,

currently, operations in the FHE domain consume significant time. We must have an efficient model with low inference times and high accuracy. We achieve this with the help of Knowledge Distillation as in Fig. 4.4.

In this section, we provide an in-depth exploration of these adaptations to allow seamless operation in the FHE domain:

1. Input adaptation
2. Model Compression via Knowledge Distillation
3. 2-D strided Convolution
4. ReLU activation function
5. Dense Layer
6. TanH activation function
7. OpenAI Gym Library

4.4.1 Input adaptation

The drone's input comprises of three consecutive images, each captured from the AirSim simulator, with dimensions 50x50. These images are concatenated to form a single input image with dimensions 50x150. In HEAAN, we adopt a strategy where each row of the image is encrypted as a single ciphertext. This approach enables the utilization of SIMD operations, enhancing computational efficiency [62].

Given that HEAAN exclusively supports the encryption of data with sizes as powers of 2, we address this constraint by padding each row of the image with zeros, extending the width to 256 [18]. Consequently, the padded input image, now of size 50x256, is encrypted, resulting in a vector of ciphertexts. To facilitate efficient computation, the plaintext weights or filters undergo similar zero-padding, aligning with the dimensions of the padded input

image. Importantly, the increase in input size from 50x150 to 50x256 does not impose a significant computational overhead, thanks to the SIMD nature of operations inherent in HEAAN.

4.4.2 Knowledge Distillation

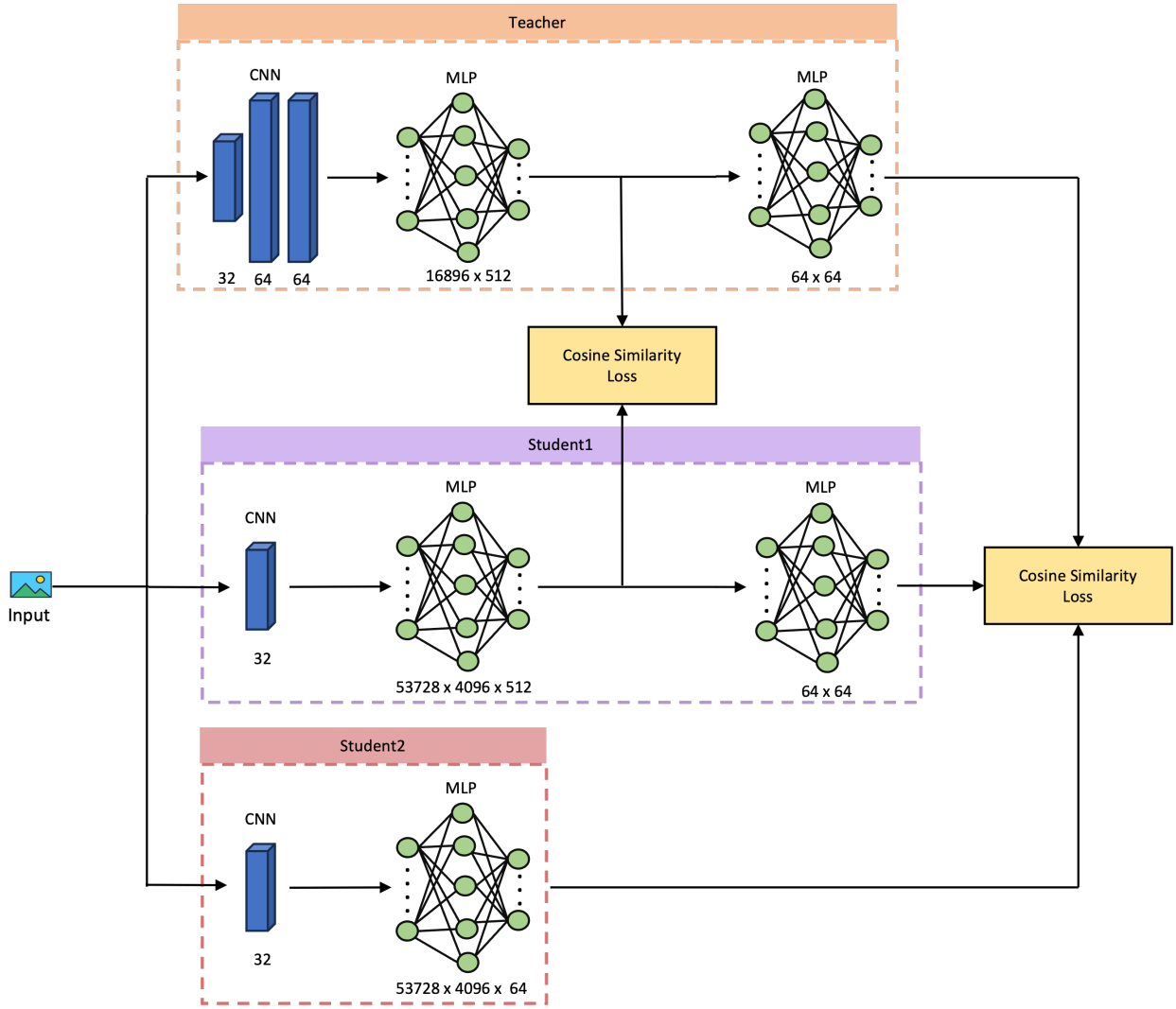


Figure 4.4: We propose a smaller model through Knowledge Distillation to suit FHE needs while maintaining privacy, security and accuracy.

Knowledge distillation, a representative type of model compression and acceleration, effectively learns a small student model from a large teacher model [63]. In our work, we employ a feature-based knowledge distillation to compress our original model (Teacher

network) to a smaller and FHE-friendly model (Student2 network). We achieve this in 2 steps as shown in Fig 4.4, achieving Student1 network first and then using Student1 to further compress the model to Student2. It is important to note that, we perform distillation only on the feature extractor network of the Teacher while training Student1. As shown in Figure 4.4, we train the student networks on the Cosine Similarity Loss between the extracted features. This significantly reduces the inference time, thereby making the FHE implementation more feasible. Ablation study as in Fig. 4.6 indicates the effect of compressing the feature extractor to a 1-layer CNN with different filter counts on the entire RL model.

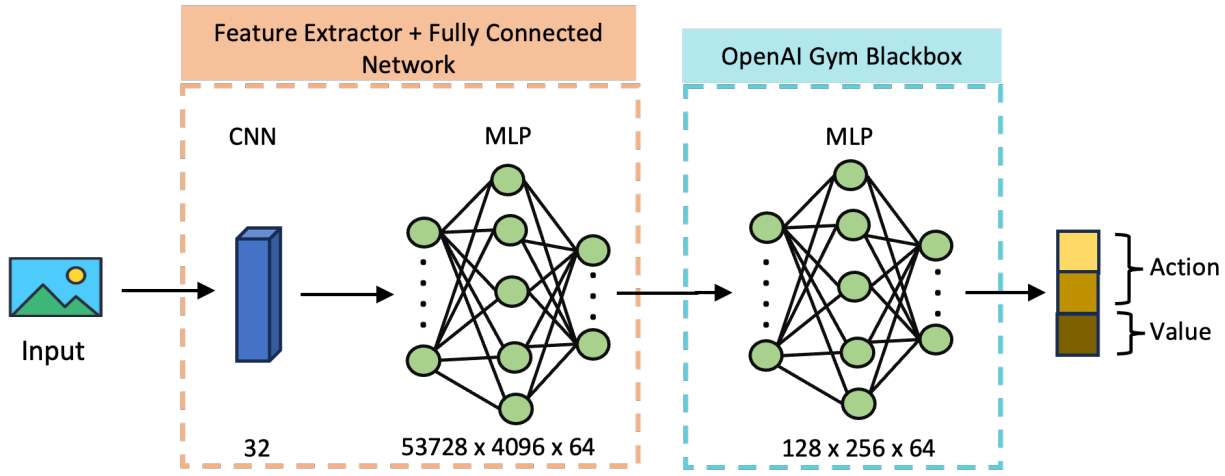


Figure 4.5: Architecture of the final compressed model (Student2 Network) to comply with FHE's time constraints.

4.4.3 Convolutional Layer

Performing regular convolution in the encrypted domain is computationally inefficient. In our research, we take a frequency-domain approach for convolution leveraging the Discrete Fourier transform (DFT) as done in [18]. The DFT of encrypted data is performed using Homomorphic Fourier transform (HFT) - inspired by Cooley-Tukey matrix factorization [64].

The following steps are performed to achieve 2D convolution efficiently:

1. HFT on each ciphertext (representative of each row in the image) as in [65]

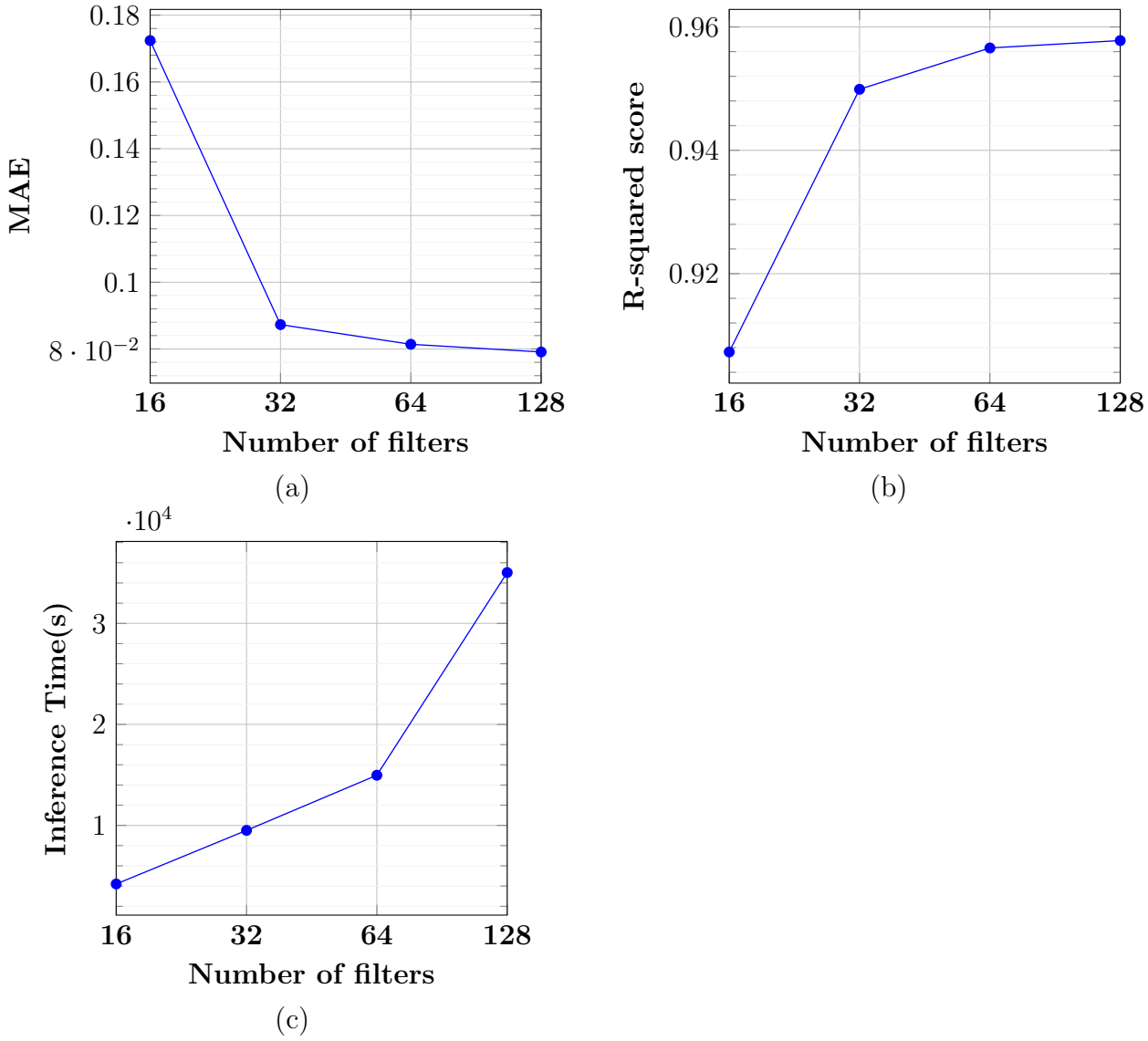


Figure 4.6: (a) Mean Absolute Error (MAE) for various filter counts in the feature-extractor of the Student network (b) R-squared score for various filter counts in the feature-extractor of the Student network (c) Inference time in seconds for various filter counts in the feature-extractor of the Student network.

2. Transpose the result based on the method in [66]

3. Perform HFT again on the transposed ciphertexts

4. Transpose the ciphertexts again

5. Compute the convolution output $y[n]$ using element-wise multiplication in the frequency domain and Inverse DFT (Inverse HFT in the encrypted domain), as expressed

in Equation 4.1.

$\mathcal{G}^{-1}$ denotes the Inverse 2D DFT, and $H[u,v]$ and $F[u,v]$ are the 2D DFT of the ciphertext and filter, respectively.

$$y[m,n] = \mathcal{G}^{-1}\left\{H[u,v] \cdot F[u,v]\right\} \tag{4.1}$$

$$H[u,v] = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} h[m,n] \cdot e^{-j\frac{2\pi}{M}um} \cdot e^{-j\frac{2\pi}{N}vn} \tag{4.2}$$

To achieve convolution with stride, a rotational manipulation is applied to the resulting ciphertext after regular convolution. We apply left rotation on the resulting ciphertext by $(N - (2 * padding))\%N$ and down rotation by $2 * padding$, where N represents the size of the ciphertext and *padding* represents the padded value used to extract DFT convolution output. Then, this result is multiplied by an array containing 1s and 0s to obtain appropriate convolution based on the stride value, as illustrated in Fig. 4.7.

4.4.4 Activation functions

Activation functions play a crucial role in neural networks, but their implementation in the context of FHE presents unique challenges [67]. FHE libraries lack native support for comparison operations, necessitating the use of approximations like CompG for the sign function [68]. Normalization is essential to align input values within the required range, achieved by scaling the outputs of convolutional layers based on the maximum observed absolute values during training. This scaling factor is determined by the maximum of the absolute values of the inputs' observed range. Following the application of the approximations, positive input values are rescaled to their original range using the inverse of the scaling factor.

In our research, we adopt a composite approximation technique for comparison in ReLU implementation from [69]. This method evaluates the input value a against zero, encoding the output as 1 for $a > 0$, 0 for $a < 0$, and 0.5 for $a = 0$, and subsequently calculates the

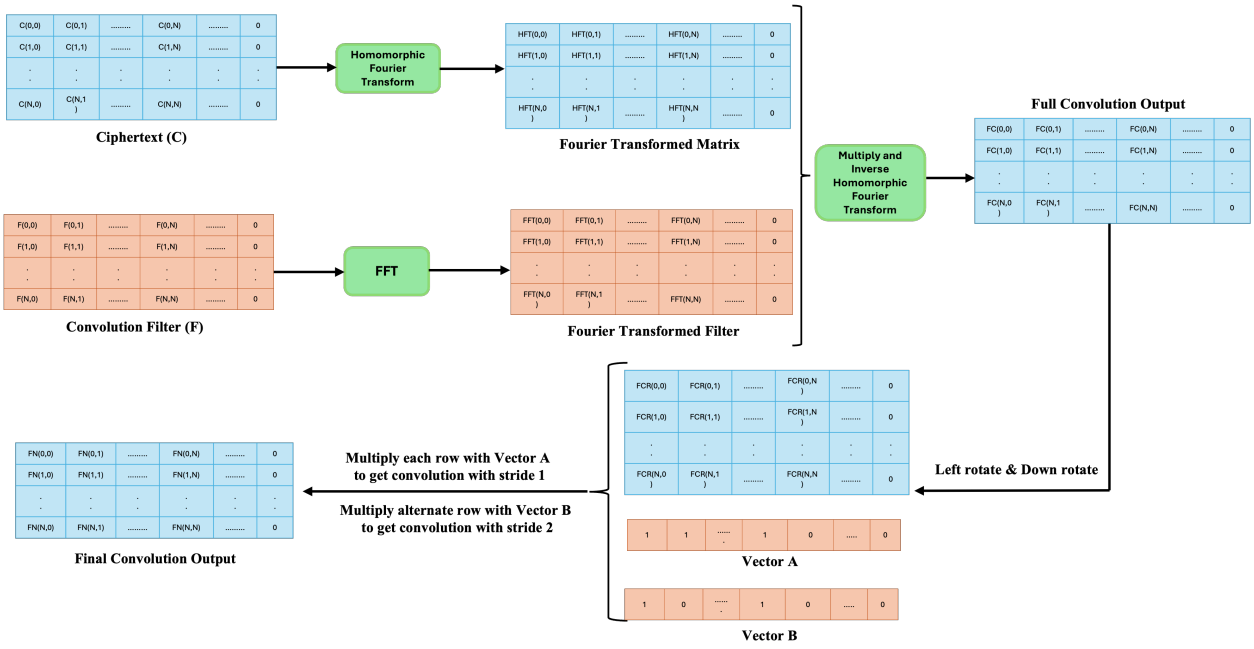
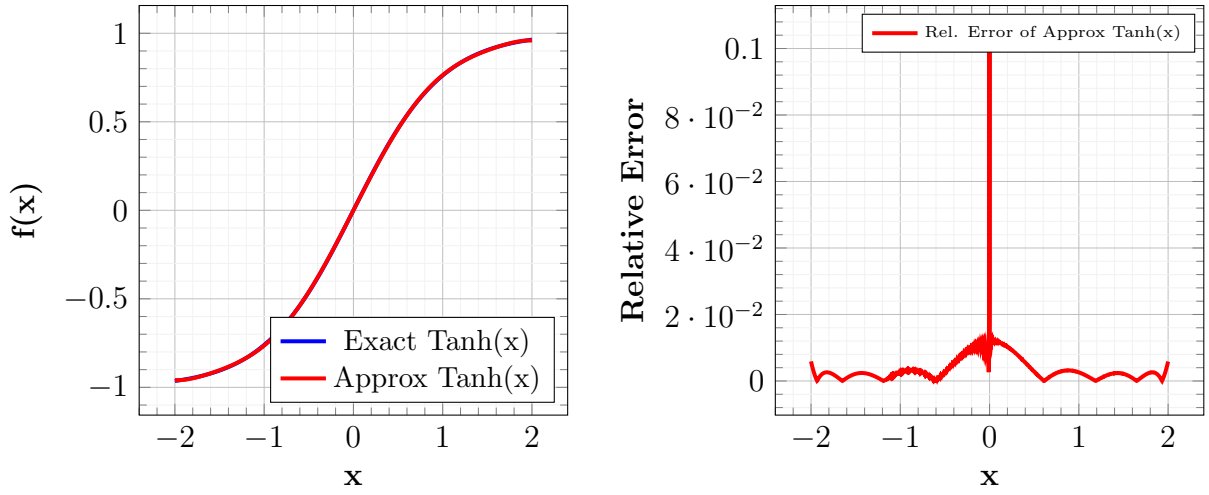


Figure 4.7: 2D Convolution in FHE Domain. Input ciphertext and weights are multiplied in the frequency domain to obtain full convolution. Final convolution output is obtained by rotating the full convolution as shown above. Different stride-based convolutions can be extracted by multiplying appropriate vectors.



(a) (b)

Figure 4.8: (a) Polynomial approximation of Tanh(x) vs Exact Tanh(x) (b) Relative error $\frac{|f(x)-tanh(x)|}{|tanh(x)|}$ of the polynomial approximation f(x) over the interval [-2, 2].

final ReLU output by multiplying this result by the input value a. Additionally, we address the challenges of implementing exponential functions in FHE by employing an 8-degree

polynomial approximation of TanH restricted to the range [-2, 2]. This approach allows for a closer approximation while mitigating the limitations of FHE in handling exponential functions. The performance of our approximation is evaluated through the relative error of 2000 points within the specified range, providing insights into its effectiveness and accuracy as shown in Fig 4.8.

4.4.5 Flattening layer

The flattening operation is usually performed on the convolution outputs. Flattening operation is not possible in FHE without decrypting and re-encrypting the ciphertexts, as it involves changing the length of ciphertexts. To circumvent this issue, we perform element-wise multiplication of the weights and convolution output. Element-wise multiplication is an extremely time-consuming operation as it involves multiplication, addition, and left rotation. We multiply each ciphertext with its corresponding weight vector and add it to a temporary ciphertext initialized to zeros. Then, we perform a summation of the ciphertext elements through repetitive left rotation and addition N-1 times.

4.4.6 Fully-Connected Layer

A Fully Connected Layer is adapted to FHE as the matrix multiplication of ciphertext inputs and plaintext weight matrices. Each row of weight matrix is multiplied with the ciphertext and the elements of the ciphertext are summed through left rotation.

4.4.7 OpenAI Gym Library

We have adapted the OpenAI Gym Library to FHE through a 3-layer neural network as in Figure 4.2 and Figure 4.5. This is due to the limitations of FHE in modeling probability distributions. The neural network learns the probability distribution and maps the final 64-dimension latent vector to the action output. The model is trained in the unencrypted

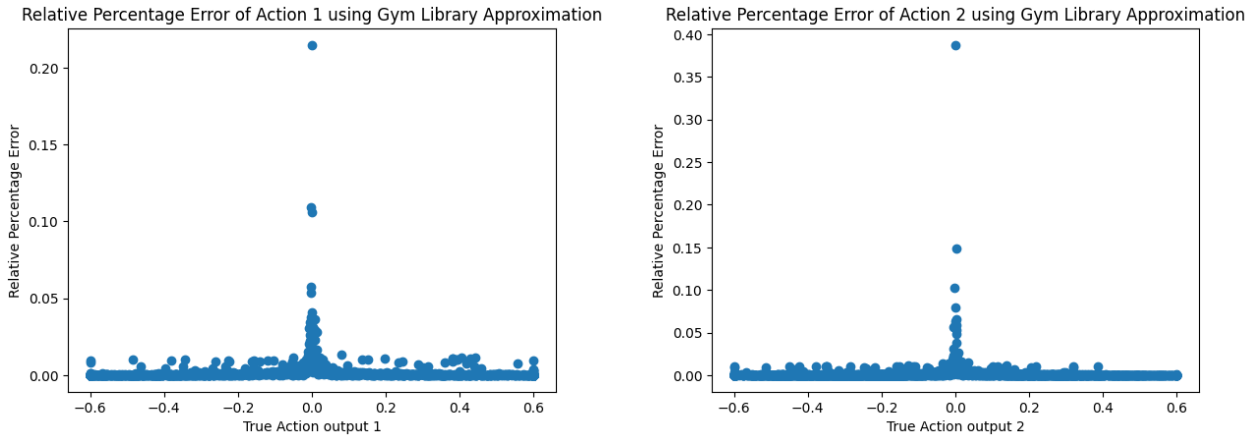
domain and its weights are used for inferencing in FHE.

4.5 Results

Experiments were performed in the encrypted domain on a subset of randomly selected samples from the testing set of the unencrypted domain. We evaluated our results from the FHE-adapted Reinforcement Learning framework against the expected results from the Reinforcement Learning framework in the unencrypted domain. Table 4.1 depicts the mean absolute error (MAE) across each block in the Teacher and Student networks within the encrypted domain. Crucially, the regression-based prediction output remained consistent between the FHE version and the plaintext counterpart for the tested samples, indicating coherence in predictive outcomes. We have also achieved an **R-squared score of 0.9631 for the Teacher network** and **0.9499 for the Student2 network** with the end-to-end FHE-based Reinforcement Learning framework, in comparison with results in the unencrypted domain. Additionally, Table 4.2 presents the average processing time across each block in the Teacher and Student networks. We achieve an 18x improvement in inference speed with Knowledge Distillation. These findings substantiate the efficacy of our FHE-adapted network, showcasing the viability of FHE in preserving model accuracy while ensuring data confidentiality.

Table 4.1: Layerwise average Mean Absolute Error (MAE) between plain-text and FHE model intermediate outputs in Teacher and Student networks.

Layer	**Avergae MAE**		
	Teacher	**Student1**	**Student2**
Convolution	0.0779	0.0860	0.0873
Linear	0.0129	0.0185	0.0203
OpenAI Gym Library Blackbox	0.0210	0.0206	0.0201



(a) (b)

Figure 4.9: Relative percentage errors of actions on adaption of OpenAI Gym Library to FHE.

Table 4.2: Time taken by the Teacher and Student networks.

Layer	**Inference Time (seconds)**		
	Teacher	**Student1**	**Student2**
Convolution	1,006,337.18	9,508.44	9,510.22
Linear	13,662.48	43,670.76	41,989.52
OpenAI Gym Library Blackbox	4,574.82	4,725.92	4,668.19
Total	1,024,754.48	57,905.12	56,167.93

Chapter5

Conclusion

Through two real-world applications in face analytics and autonomous UAVs, we underscore the usefulness of FHE in providing strict privacy and security without losing performance. Through our experiments in both cases, we have shown that FHE is the way forward in fostering the privacy and security aspects of Trustworthy AI.

Face Analytics. In this report, we propose to use FHE in combination with template protection and compression to secure the face template and prevent soft biometric leakage. We show that soft biometric attributes from face embeddings can be strictly protected while preserving identification accuracy. In our approach, we compress the face embeddings using MRL (Matryoksha Representation Learning), encrypt them, and then apply PolyProtect as the template protection scheme. The identification performance of the encrypted template compared with the unencrypted version is unchanged. Since FHE guarantees are based on strong theoretical principles, privacy and security are ensured, and only authorized individuals with the secret key will be able to access the results from the FHE computation.

Autonomous UAVs. We adopt a groundbreaking end-to-end homomorphically encrypted Unmanned Aerial Vehicle (UAV) navigation system, that uses a fusion of reinforcement learning and Fully Homomorphic Encryption (FHE) from [**aggarwal2024enhancing**]. Given the model's high latency, we propose a model compression technique to significant speedup (18x) inference time. We achieve this Knowledge Distillation in 2 steps, using the cosine distance metric to train the student models. First, we compress the 3 layer CNN to a

1 layer CNN, achieving Student1. Then, we compress the MLP layer using cosine loss from both Student1 and Teacher, to achieve our compressed model in Student2. In addition, we provide detailed steps to implement the entire model architecture in FHE. In our evaluation of inference, our proposed FHE-based compressed architecture demonstrates lower latency with minimal error across each block in the network, showcasing no discernible accuracy loss when compared to its plaintext counterpart.

Chapter6

Relevant Publications

1. Enhancing Privacy in Face Analytics Using Fully Homomorphic Encryption (*IEEE International Conference on Automatic Face and Gesture Recognition 2024*)

2. Enhancing Privacy and Security of Autonomous UAV Navigation (*IEEE Conference on Artificial Intelligence 2024*)

Bibliography

[1] Anil K. Jain, Arun Ross, and Karthik Nandakumar. *Introduction to Biometrics: A Textbook*. Springer Science & Business Media, 2011.

[2] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. "DeepFace: Closing the Gap to Human-Level Performance in Face Verification". In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2014.

[3] 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 (CVPR)*. 2022, pp. 18750–18759.

[4] Stan Z. Li, Anil K. Jain, and Jiankang Deng, eds. *Handbook of Face Recognition*. Springer Cham, 2024.

[5] Anil K. Jain, Karthik Nandakumar, and Arun Ross. "50 Years of Biometric Research: Accomplishments, Challenges, and Opportunities". In: *Pattern Recognition Letters* 79.C (2016), 80–105.

[6] Antitza Dantcheva, Petros Elia, and Arun Ross. "What Else Does Your Biometric Data Reveal? A Survey on Soft Biometrics". In: *IEEE Transactions on Information Forensics and Security* 11.3 (2016), pp. 441–467.

[7] Denise Almeida, Konstantin Shmarko, and Elizabeth Lomas. "The ethics of facial recognition technologies, surveillance, and accountability in an age of artificial intelligence: a comparative analysis of US, EU, and UK regulatory frameworks". In: *AI and Ethics* 2.3 (2022), pp. 377–387.

[8] `https://tinyurl.com/4p2af252`. 2023.

[9] `https://tinyurl.com/3z7ceksu`. 2023.

[10] Bryce Westlake, Russell Brewer, Thomas Swearingen, Arun Ross, Stephen Patterson, Dana Michalski, Martyn Hole, Katie Logos, Richard Frank, David Bright, and Erin Afana. "Developing automated methods to detect and match face and voice biometrics in child sexual abuse videos". In: *Trends and Issues in Crime and Criminal Justice* 648 (2022), pp. 1–15.

[11] Syed Agha Hassnain Mohsan, Muhammad Asghar Khan, Fazal Noor, Insaf Ullah, and Mohammed H. Alsharif. "Towards the Unmanned Aerial Vehicles (UAVs): A Comprehensive Review". In: *Drones* 6.6 (2022). ISSN: 2504-446X. DOI: `10.3390/drones6060147`. URL: `https://www.mdpi.com/2504-446X/6/6/147`.

[12] Gui-Song Xia Yuncheng Lu Zhucun Xue and Liangpei Zhang. "A survey on vision-based UAV navigation". In: *Geo-spatial Information Science* 21.1 (2018), pp. 21–32. DOI: 10.1080/10095020.2017.1420509. eprint: https://doi.org/10.1080/10095020.2017.1420509. URL: https://doi.org/10.1080/10095020.2017.1420509.

[13] Chao Wang, Jian Wang, Jingjing Wang, and Xudong Zhang. "Deep-Reinforcement-Learning-Based Autonomous UAV Navigation With Sparse Rewards". In: *IEEE Internet of Things Journal* 7.7 (2020), pp. 6180–6190. DOI: 10.1109/JIOT.2020.2973193.

[14] Chao Wang, Jian Wang, Yuan Shen, and Xudong Zhang. "Autonomous Navigation of UAVs in Large-Scale Complex Environments: A Deep Reinforcement Learning Approach". In: *IEEE Transactions on Vehicular Technology* 68.3 (2019), pp. 2124–2136. DOI: 10.1109/TVT.2018.2890773.

[15] Sifat Rezwan and Wooyeol Choi. "Artificial Intelligence Approaches for UAV Navigation: Recent Advances and Future Challenges". In: *IEEE Access* 10 (2022), pp. 26320–26339. DOI: 10.1109/ACCESS.2022.3157626.

[16] Yassine Mekdad, Ahmet Aris, Leonardo Babun, Abdeslam El Fergougui, Mauro Conti, Riccardo Lazzeretti, and A Selcuk Uluagac. "A survey on security and privacy issues of UAVs". In: *Computer networks* 224 (2023). ISSN: 1389-1286.

[17] Rongxiao Guo, Buhong Wang, and Jiang Weng. "Vulnerabilities and Attacks of UAV Cyber Physical Systems". In: *Proceedings of the International Conference on Computing, Networks and Internet of Things*. Association for Computing Machinery, 2020, 8–12. ISBN: 9781450377713. DOI: 10.1145/3398329.3398331. URL: https://doi.org/10.1145/3398329.3398331.

[18] Vatsal Aggarwal, Arjun Ramesh Kaushik, and Nalini Ratha. "Enhancing Privacy and Security of Autonomous UAV Navigation". In: *Conference on Artificial Intelligence*. 2024.

[19] Ronald L. Rivest, Leonard M. Adleman, and Michael L. Dertouzos. "On data banks and privacy homomorphisms". In: *Foundations of secure computation* 4 (1978), pp. 169–180.

[20] Kundan Munjal and Rekha Bhatia. "A systematic review of homomorphic encryption and its contributions in healthcare industry". In: *Complex & Intelligent Systems* 9 (May 2022), pp. 1–28. DOI: 10.1007/s40747-022-00756-z.

[21] Shruthi Gorantala, Rob Springer, and Bryant Gipson. "Unlocking the Potential of Fully Homomorphic Encryption". In: *Commun. ACM* 66.5 (2023), 72–81. ISSN: 0001-0782. DOI: 10.1145/3572832. URL: https://doi.org/10.1145/3572832.

[22] Shai Halevi and Victor Shoup. *Design and implementation of HElib: a homomorphic encryption library*. Cryptology ePrint Archive, Paper. 2020. URL: https://eprint.iacr.org/2020/1481.

[23] *Microsoft SEAL (release 4.1)*. https://github.com/Microsoft/SEAL. Microsoft Research, Redmond, WA. Jan. 2023.

[24] *PALISADE Homomorphic Encryption Software Library*. URL: https://palisade-crypto.org/.

[25] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. *TFHE: Fast Fully Homomorphic Encryption Library*. https://tfhe.github.io/tfhe/. August 2016.

[26] Tanping Zhou, Xiaoyuan Yang, Longfei Liu, Wei Zhang, and Ningbo Li. "Faster Bootstrapping With Multiple Addends". In: *IEEE Access* 6 (2018), pp. 49868–49876. DOI: 10.1109/ACCESS.2018.2867655.

[27] Clément Hoffmann, Pierrick Méaux, and Thomas Ricosset. *Transciphering, using FiLIP and TFHE for an efficient delegation of computation*. Cryptology ePrint Archive, Paper. 2020. URL: https://eprint.iacr.org/2020/1373.

[28] Jung Hee Cheon, Andrey Kim, Miran Kim, and Yongsoo Song. "Homomorphic Encryption for Arithmetic of Approximate Numbers". In: *Advances in Cryptology – ASIACRYPT*. Ed. by Tsuyoshi Takagi and Thomas Peyrin. Cham: Springer International Publishing, 2017, pp. 409–437.

[29] Vedrana Krivokuća Hahn and Sébastien Marcel. "Towards Protecting Face Embeddings in Mobile Face Verification Scenarios". In: *IEEE Transactions on Biometrics, Behavior, and Identity Science* 4.1 (2022), pp. 117–134. DOI: 10.1109/TBIOM.2022.3140472.

[30] Baiyu Li and Daniele Micciancio. "On the Security of Homomorphic Encryption on Approximate Numbers". In: *Advances in Cryptology – EUROCRYPT: 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17–21, Proceedings, Part I*. Springer-Verlag, 2021, 648–677. ISBN: 978-3-030-77869-9. DOI: 10.1007/978-3-030-77870-5_23. URL: https://doi.org/10.1007/978-3-030-77870-5_23.

[31] Arun Ross, Sudipta Banerjee, and Anurag Chowdhury. "Deducing health cues from biometric data". In: *Computer Vision and Image Understanding* 221 (2022), p. 103438.

[32] Blaž Bortolato, Marija Ivanovska, Peter Rot, Janez Križaj, Philipp Terhörst, Naser Damer, Peter Peer, and Vitomir Štruc. "Learning privacy-enhancing face representations through feature disentanglement". In: *15th IEEE International Conference on*

Automatic Face and Gesture Recognition. 2020, pp. 495–502. DOI: 10.1109/FG47880.2020.00007.

[33] Aythami Morales, Julian Fierrez, Ruben Vera-Rodriguez, and Ruben Tolosana. "SensitiveNets: Learning Agnostic Representations with Application to Face Images". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.6 (2021), pp. 2158–2164. DOI: 10.1109/TPAMI.2020.3015420.

[34] Pietro Melzi, Hatef Otroshi Shahreza, Christian Rathgeb, Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, Sébastien Marcel, and Christoph Busch. "Multi-IVE: Privacy Enhancement of Multiple Soft-Biometrics in Face Embeddings". In: *IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*. 2023, pp. 323–331. DOI: 10.1109/WACVW58289.2023.00036.

[35] Zohra Rezgui, Nicola Strisciuglio, and Raymond Veldhuis. "Enhancing Soft Biometric Face Template Privacy With Mutual Information-Based Image Attacks". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*. 2024, pp. 1141–1149.

[36] Yushu Zhang, Tao Wang, Ruoyu Zhao, Wenying Wen, and Youwen Zhu. "RAPP: Reversible Privacy Preservation for Various Face Attributes". In: *IEEE Transactions on Information Forensics and Security* 18 (2023), pp. 3074–3087. DOI: 10.1109/TIFS.2023.3274359.

[37] Vahid Mirjalili, Sebastian Raschka, and Arun Ross. "PrivacyNet: Semi-Adversarial Networks for Multi-Attribute Face Privacy". In: *IEEE Transactions on Image Processing* 29 (2020), pp. 9400–9412. DOI: 10.1109/TIP.2020.3024026.

[38] Vishnu Boddeti. "Secure Face Matching Using Fully Homomorphic Encryption". In: *IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. 2018, pp. 1–10. DOI: 10.1109/BTAS.2018.8698601.

[39] Joshua J. Engelsma, Anil K. Jain, and Vishnu Naresh Boddeti. "HERS: Homomorphically Encrypted Representation Search". In: *IEEE Transactions on Biometrics, Behavior, and Identity Science* 4.3 (2022), pp. 349–360. DOI: 10.1109/TBIOM.2021.3139866.

[40] 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: *IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. 2020, pp. 1127–1134. DOI: 10.1109/TrustCom50675.2020.00149.

[41] Aditya Malik, Nalini Ratha, Bharat Yalavarthi, Tilak Sharma, Arjun Kaushik, and Charanjit Jutla. "Confidential and Protected Disease Classifier using Fully Homomorphic Encryption". In: 2024. arXiv: 2405.02790 [cs.CR].

[42] Inioluwa Deborah Raji, Timnit Gebru, Margaret Mitchell, Joy Buolamwini, Joonseok Lee, and Emily Denton. "Saving Face: Investigating the Ethical Concerns of Facial Recognition Auditing". In: *CoRR* abs/2001.00964 (2020). arXiv: 2001.00964. URL: http://arxiv.org/abs/2001.00964.

[43] Joseph Robinson. *Balanced Faces in the Wild*. IEEE Dataport. 2022. URL: https://dx.doi.org/10.21227/nmsj-df12.

[44] Florian Schroff, Dmitry Kalenichenko, and James Philbin. "FaceNet: A Unified Embedding for Face Recognition and Clustering". In: *CoRR* abs/1503.03832 (2015). arXiv: 1503.03832. URL: http://arxiv.org/abs/1503.03832.

[45] Minchul Kim, Anil K. Jain, and Xiaoming Liu. "AdaFace: Quality Adaptive Margin for Face Recognition". In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 18729–18738. DOI: 10.1109/CVPR52688.2022.01819.

[46] Vedrana Krivokuća Hahn and Sébastien Marcel. "Biometric Template Protection for Neural-Network-Based Face Recognition Systems: A Survey of Methods and Evaluation Techniques". In: *IEEE Transactions on Information Forensics and Security* 18 (2023), pp. 639–666. DOI: 10.1109/TIFS.2022.3228494.

[47] Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham Kakade, Prateek Jain, and Ali Farhadi. "Matryoshka Representation Learning". In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh. Vol. 35. Curran Associates, Inc., 2022, pp. 30233–30249. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/c32319f4868da7613d78af9993100e42-Paper-Conference.pdf.

[48] Jean-Paul Yaacoub, Hassan Noura, Ola Salman, and Ali Chehab. "Security analysis of drones systems: Attacks, limitations, and recommendations". In: *Internet of Things* 11 (2020), p. 100218. ISSN: 2542-6605. DOI: https://doi.org/10.1016/j.iot.2020.100218. URL: https://www.sciencedirect.com/science/article/pii/S2542660519302112.

[49] Maria Rigaki and Sebastian Garcia. "A Survey of Privacy Attacks in Machine Learning". In: *ACM Comput. Surv.* 56.4 (2023). ISSN: 0360-0300. DOI: 10.1145/3624010. URL: https://doi.org/10.1145/3624010.

[50] Bilal Kabas. "Autonomous UAV Navigation via Deep Reinforcement Learning Using PPO". In: *30th Signal Processing and Communications Applications Conference (SIU)*. 2022, pp. 1–4. DOI: 10.1109/SIU55565.2022.9864769.

[51] Wencheng Yang, Song Wang, Xuefei Yin, Xu Wang, and Jiankun Hu. "A Review on Security Issues and Solutions of the Internet of Drones". In: *IEEE Open Journal of the Computer Society* 3 (2022), pp. 96–110. DOI: 10.1109/OJCS.2022.3183003.

[52] Vikas Hassija, Vinay Chamola, Adhar Agrawal, Adit Goyal, Nguyen Cong Luong, Dusit Niyato, Fei Richard Yu, and Mohsen Guizani. "Fast, Reliable, and Secure Drone Communication: A Comprehensive Survey". In: *IEEE Communications Surveys & Tutorials* 23.4 (2021), pp. 2802–2832. DOI: 10.1109/COMST.2021.3097916.

[53] C. G. Leela Krishna and Robin R. Murphy. "A review on cybersecurity vulnerabilities for unmanned aerial vehicles". In: *IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*. 2017, pp. 194–199. DOI: 10.1109/SSRR.2017.8088163.

[54] Mohammed Y. Alzahrani, Nayeem Ahmad Khan, Lilia Georgieva, Alawi M. Bamahdi, Omar Ahmed Abdulkader, and Ahmed H. Alahmadi. "Protecting Attacks on Unmanned Aerial Vehicles using Homomorphic Encryption". English. In: *Indonesian Journal of Electrical Engineering and Informatics* 11.1 (Mar. 2023). Publisher Copyright: © Institute of Advanced Engineering and Science. All rights reserved., pp. 88–96. ISSN: 2089-3272. DOI: 10.52549/ijeei.v11i1.3932.

[55] Jung Hee Cheon, Kyoohyung Han, Seong-Min Hong, Hyoun Jin Kim, Junsoo Kim, Suseong Kim, Hosung Seo, Hyungbo Shim, and Yongsoo Song. "Toward a Secure Drone System: Flying With Real-Time Homomorphic Authenticated Encryption". In: *IEEE Access* 6 (2018), pp. 24325–24339. DOI: 10.1109/ACCESS.2018.2819189.

[56] Tianyuan Liu, Hongpeng Guo, Claudiu Danilov, and Klara Nahrstedt. "A Privacy-Preserving Data Collection and Processing Framework for Third-Party UAV Services". In: *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. 2020, pp. 683–690. DOI: 10.1109/TrustCom50675.2020.00095.

[57] Vikas Hassija, Vinay Chamola, Adhar Agrawal, Adit Goyal, Nguyen Cong Luong, Dusit Niyato, Fei Richard Yu, and Mohsen Guizani. "Fast, Reliable, and Secure Drone Communication: A Comprehensive Survey". In: vol. 23. 4. 2021, pp. 2802–2832. DOI: 10.1109/COMST.2021.3097916.

[58] Jung Hee Cheon, Kyoohyung Han, Seong-Min Hong, Hyoun Jin Kim, Junsoo Kim, Suseong Kim, Hosung Seo, Hyungbo Shim, and Yongsoo Song. "Toward a Secure Drone System: Flying With Real-Time Homomorphic Authenticated Encryption". In: vol. 6. 2018, pp. 24325–24339. DOI: 10.1109/ACCESS.2018.2819189.

[59] Wei Ao and Vishnu Naresh Boddeti. "AutoFHE: Automated Adaption of CNNs for Efficient Evaluation over FHE". In: https://eprint.iacr.org/2023/162. 2023. URL: https://eprint.iacr.org/2023/162.

[60] Yang He and Lingao Xiao. "Structured Pruning for Deep Convolutional Neural Networks: A Survey". In: vol. 46. 5. 2024, pp. 2900–2919. DOI: 10.1109/TPAMI.2023.3334614.

[61] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. "Proximal Policy Optimization Algorithms." In: *CoRR* abs/1707.06347 (2017). URL: http://dblp.uni-trier.de/db/journals/corr/corr1707.html#SchulmanWDRK17.

[62] Wonkyung Jung, Eojin Lee, Sangpyo Kim, Jongmin Kim, Namhoon Kim, Keewoo Lee, Chohong Min, Jung Hee Cheon, and Jung Ho Ahn. "Accelerating Fully Homomorphic Encryption Through Architecture-Centric Analysis and Optimization". In: *IEEE Access* 9 (2021), pp. 98772–98789. DOI: 10.1109/ACCESS.2021.3096189.

[63] Jianping Gou, Baosheng Yu, Stephen J. Maybank, and Dacheng Tao. "Knowledge Distillation: A Survey". In: vol. 129. 6. Springer Science and Business Media LLC, Mar. 2021, 1789–1819. DOI: 10.1007/s11263-021-01453-z. URL: http://dx.doi.org/10.1007/s11263-021-01453-z.

[64] James W. Cooley and John W. Tukey. "An Algorithm for the Machine Calculation of Complex Fourier Series". In: *Mathematics of Computation* 19.90 (1965), pp. 297–301. ISSN: 00255718, 10886842. URL: http://www.jstor.org/stable/2003354 (visited on 01/06/2024).

[65] Kyoohyung Han, Minki Hhan, and Jung Hee Cheon. "Improved Homomorphic Discrete Fourier Transforms and FHE Bootstrapping". In: *IEEE Access* 7 (2019), pp. 57361–57370. DOI: 10.1109/ACCESS.2019.2913850.

[66] Ahmed Zekri. "Enhancing the Matrix Transpose Operation Using Intel Avx Instruction Set Extension". In: *International Journal of Computer Science & Information Technology* 6 (June 2014), pp. 67–78. DOI: 10.5121/ijcsit.2014.6305.

[67] Jung Hee Cheon, Dongwoo Kim, Duhyeong Kim, Hun Hee Lee, and Keewoo Lee. "Numerical Method for Comparison on Homomorphically Encrypted Numbers". In: 2019.

[68] Jung Hee Cheon, Dongwoo Kim, and Duhyeong Kim. "Efficient Homomorphic Comparison Methods with Optimal Complexity". In: 2019.

[69] Tilak Sharma, Mahika Wason, Vishnu Boddeti, Arun Ross, and Nalini Ratha. "Fully Homomorphic Encryption Operators for Score and Decision Fusion in Biometric Identification". In: *IEEE International Workshop on Information Forensics and Security (WIFS)*. 2023, pp. 1–6. DOI: 10.1109/WIFS58808.2023.10374571.