# Privacy-Preserving Biometric Template Protection Using Homomorphic Encryption

**Orel Razy, ID: XXXX**

**Supervised by:**
This project was supervised by Prof. Adi Akavia, Fall 2024

## Abstract

Biometric identification systems have become pervasive across multiple industries, raising critical privacy concerns due to potential data breaches. This project addresses these concerns by implementing privacy-preserving biometric identification through Fully Homomorphic Encryption (FHE) using the GhostFaceNet model and the MS1M-ArcFace dataset. We investigate how the inherent precision limitations of the CKKS encryption scheme impact the accuracy of biometric matching. Our findings show negligible differences between cleartext and encrypted computations, with over 96% accuracy in top-10 matches and compact ciphertext sizes of under 11 MB, demonstrating the feasibility and efficiency of our approach.

# 1 Introduction

## 1.1 Background

Biometric identification leverages unique physiological traits, such as facial features, to authenticate individuals. While this technology has revolutionized security and authentication, it also presents significant privacy risks due to potential leaks of sensitive biometric data. Fully Homomorphic Encryption (FHE) addresses these risks by enabling computations directly on encrypted data, preserving confidentiality throughout the computation process.

## 1.2 Research Goal

The primary goal of this project is to develop a privacy-preserving biometric identification pipeline that integrates vector-based facial recognition with FHE. Specifically, we evaluate the impact of precision limitations introduced by CKKS encryption on the system's accuracy and runtime performance. Initially, the project was scoped to investigate only Part A, focusing on evaluating the effect of precision limitations on biometric identification under cleartext data. However, recognizing the potential impact of integrating privacy-preserving computations, we extended our efforts to implement Part B (similarity metric computation over encrypted vectors) and Part C (privacy-preserving biometric identification). The datasets used for Parts A and C differ due to time constraints and the exploratory nature of the later phases.

## 1.3 Results

Our implementation of privacy-preserving biometric identification demonstrates that encrypted cosine similarity computations achieve high accuracy, with an average top-10 match percentage of 96.68%. The system efficiently handles the biometric comparison of 10,000 individuals with ciphertext sizes of 10.8 MB. However, the computational overhead of 8622.54 seconds for encrypted similarity indicates room for future optimizations.

## 1.4 Related Work

Existing literature on privacy-preserving biometrics explores techniques like secure multiparty computation (SMPC) and FHE. While SMPC offers privacy guarantees, it typically requires significant communication overhead. FHE, on the other hand, allows encrypted computations on cloud servers with minimal interaction. The MS1M-ArcFace dataset and the GhostFaceNet model have been widely adopted in facial recognition studies due to their robustness and state-of-the-art accuracy.

### Comparison with Related Work

**Secure Multiparty Computation (SMPC) in Biometric Identification:**
Bringer et al. [1] presented a comprehensive overview of applying secure two-party computation techniques to biometric identification systems, highlighting significant communication overheads as a main limitation. Wu et al. [4] introduced tensor triples for multidimensional SMPC protocols, achieving over 1000x speedup but with considerable offline costs.

**Fully Homomorphic Encryption (FHE) in Biometric Identification:**
Sperling et al. [3] proposed HEFT for non-interactive end-to-end secure fusion and matching of biometric templates. Pradel and Mitchell [2] developed a privacy-preserving biometric authentication protocol with robust security guarantees, though facing high runtime overhead.

Our approach aligns with these works by relying on CKKS-based FHE to preserve privacy without interaction, maintaining accuracy above 96%. However, as with other FHE solutions, runtime overhead remains a challenge requiring further optimizations.

# 2 Technical Background / Preliminaries

**Biometric Identification:** Relies on matching vector embeddings derived from facial images using deep learning models such as GhostFaceNet.

**GhostFaceNet Model:** GhostFaceNet is an optimized, lightweight neural network for facial recognition that builds upon architectures like FaceNet and ArcFace, emphasizing efficient embedding generation with high accuracy. It generates compact yet robust 512-dimensional embeddings, suitable for large-scale comparisons.

**Similarity Metric:**
Cosine similarity is used to measure the closeness between embeddings:

$$\text{Cosine Similarity}(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|},$$

where $\mathbf{A}$ and $\mathbf{B}$ are embedding vectors and $\|\cdot\|$ denotes the Euclidean norm. We also briefly evaluated the Euclidean distance,

$$\text{Euclidean Distance}(\mathbf{A}, \mathbf{B}) = \sqrt{\sum_{i=1}^{n}(A_i - B_i)^2},$$

but it underperformed in our experiments.

**Homomorphic Encryption (CKKS):** A variant of FHE supporting approximate arithmetic computations on encrypted data. We use TenSEAL for an implementation of CKKS, carefully tuning encryption parameters to balance precision and performance.

**MS1M-ArcFace Dataset:** A large-scale facial recognition dataset containing over 5.8 million images across 85,000 identities. This ensures robust, real-world performance testing.

# 3 Results

This section outlines the experimental design, performance metrics, and results analysis. The protocol description summarizes the key steps of embedding extraction and similarity computation. The system description covers its high-level structure, a diagram reference, and implementation details. Finally, the empirical evaluation presents key performance metrics, experimental outcomes, and discusses accuracy, precision, and runtime.

## 3.1 (a) The Protocol

1. **Embedding Extraction:** Utilized the GhostFaceNet model via DeepFace to generate 512-dimensional facial embeddings from the MS1M-ArcFace dataset.

2. **Similarity Computation:** Computed the cosine similarity scores in cleartext and encrypted forms (using CKKS).

3. **Evaluation:** Assessed the accuracy across thresholds and analyzed the impact of reduced precision under FHE.

## 3.2 (b) System Description

### i. High-Level Verbal Description

Our privacy-preserving biometric system processes input images, extracts 512-dimensional facial embeddings (GhostFaceNet), and computes cosine similarity scores. We compare cleartext similarity computations to those under CKKS encryption, highlighting any performance or accuracy differences.

### ii. System Diagram

Below is a diagram illustrating the flow of data:

### iii. Implementation Details

**Libraries Used:**

- TensorFlow for deep learning operations
- DeepFace for embedding extraction
- TenSEAL for CKKS-based homomorphic encryption
- scikit-learn for performance evaluation
- matplotlib for visualization

**Encryption Scheme (CKKS):**

- poly_modulus_degree = 16384
- coefficient bit sizes = [60, 40, 40, 60]
- global scale = $2^{30}$

## 3.3 (c) Empirical Evaluation

### i. Experiments

**Hardware Configuration:**

- MacBook Pro M3, 36 GB RAM, 0.5 TB SSD

**Dataset:**

- MS1M-ArcFace, which provides diverse facial images for robust testing

**Parameter Settings:**

- Part A: 10,000 individuals, 2 images each
- Part C: 500 individuals, 3 images each

### ii. Performance

### Part A: Accuracy Across Thresholds

| Threshold | Accuracy | Precision | Recall | F1 Score |
|---|---|---|---|---|
| $1e^{-44}$ | 0.9832 | 1.0000 | 0.9832 | 0.9916 |
| $1e^{-06}$ | 0.9832 | 1.0000 | 0.9832 | 0.9916 |
| 0.05 | 0.9543 | 1.0000 | 0.9543 | 0.9766 |
| 0.2 | 0.7659 | 1.0000 | 0.7659 | 0.8674 |
| 0.5 | 0.2302 | 1.0000 | 0.2302 | 0.3742 |

Table 1: Performance metrics under varying thresholds (Part A).

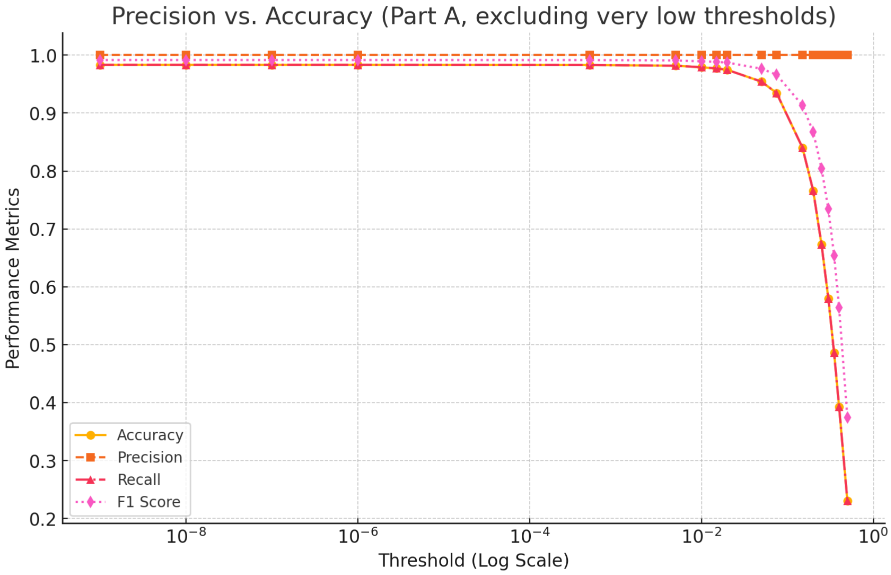

Figure 2: Variation of performance metrics (accuracy, precision, recall, F1 score) as a function of threshold in Part A.

### Part B: Similarity Metric Performance (Cleartext vs. Encrypted)

| Metric | Average | Std Dev | Max |
|---|---|---|---|
| Absolute difference | 0.0000 | 0.0000 | 0.0000 |
| Cleartext runtime (sec) | 3.63e-05 | 1.26e-05 | 1.08e-04 |
| Encryption runtime (sec) | 0.0042 | 0.0009 | 0.0110 |
| Computation runtime (sec) | 0.0073 | 0.0034 | 0.0341 |
| Total runtime (sec) | 0.0115 | 0.0039 | 0.0392 |

Table 2: Cleartext vs. Encrypted similarity computations (Part B).

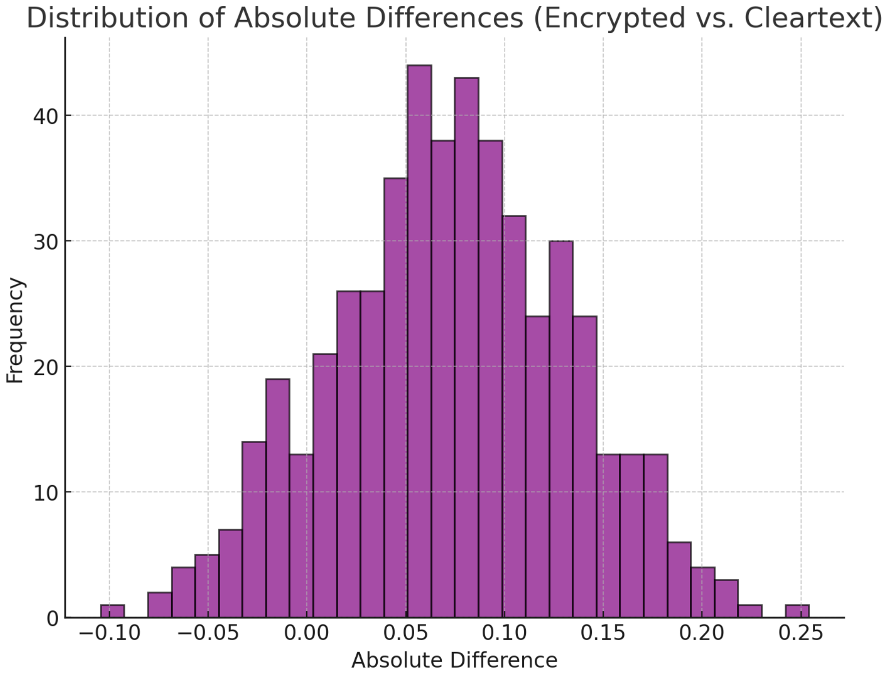

Figure 3: Distribution of absolute differences between encrypted and cleartext similarity scores (Part B).

### Part C: System Runtime and Accuracy Metrics

| Metric | Value |
|---|---|
| Embedding generation (sec) | 621.85 |
| Cleartext similarity computation | 16.03 |
| Encrypted similarity computation | 8622.54 |
| Encryption time (sec) | 7.67 |
| Average top-10 match percentage | 96.68% |
| Size of encrypted scores (MB) | 10.80 |

Table 3: Runtime and accuracy metrics (Part C).

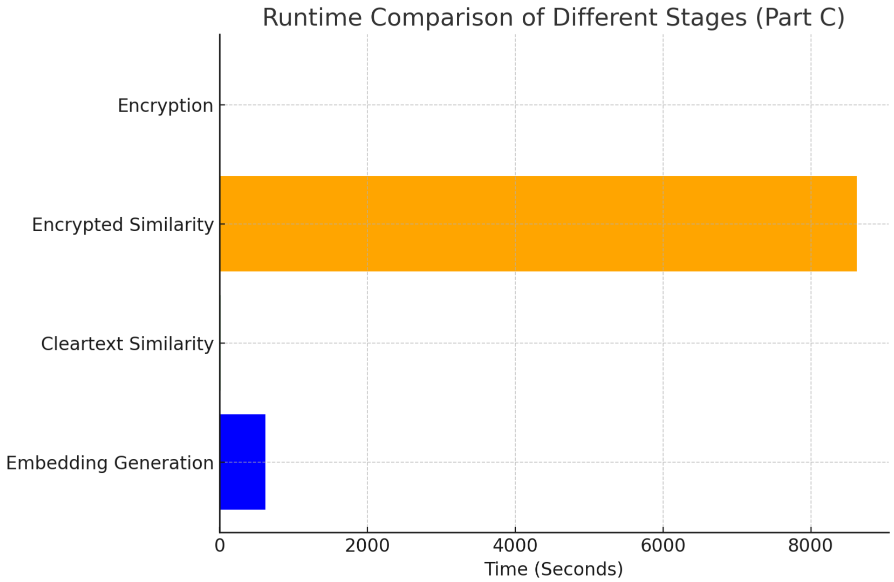

Figure 4: Comparison of runtimes for embedding generation, cleartext similarity, and encrypted similarity (Part C).

### iii. Discussion

1. **Precision and Accuracy Stability:** The system maintained high accuracy (>95%) under encryption, indicating robustness to CKKS precision constraints.
2. **Threshold Impact:** Low thresholds ($\approx 1e^{-6}$) yielded optimal accuracy; higher thresholds drastically reduced performance.
3. **Euclidean Distance Comparison:** Significantly underperformed, proving less stable under encryption noise.
4. **Runtime Overhead:** A substantial bottleneck emerged for encrypted similarity computations in Part C (8622.54 sec).
5. **Scalability:** Demonstrated feasibility for 10,000 individuals in Part A and 500 with encryption in Part C.
6. **Optimization Attempts:** SIMD packing and PCA harmed accuracy significantly, whereas batching + multithreading provided a better balance.

### iv. Nearest-Neighbor Attack Simulation

We conducted a brute-force nearest-neighbor attack to evaluate robustness against adversarial attempts. No identities were correctly inferred in either cleartext or encrypted data (0% attack success). However, in cleartext form, high-confidence similarity matches ($> 0.95$) were occasionally observed, implying potential vulnerability if combined with auxiliary data. Encrypted scores remained uniformly small, preventing any meaningful inference by adversaries.

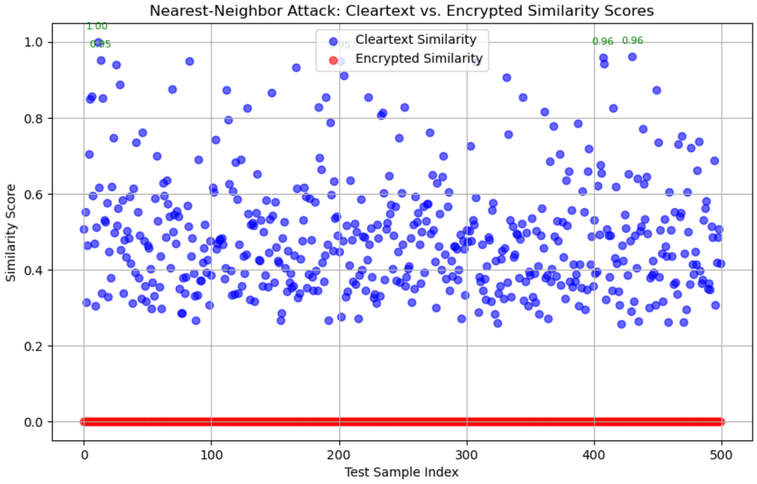

Figure 5: Comparison of nearest-neighbor attack results: Cleartext vs. Encrypted similarity scores.

# 4 Conclusions

This project successfully implemented a privacy-preserving biometric identification pipeline using Fully Homomorphic Encryption (FHE) and the GhostFaceNet model. By leveraging CKKS-based encryption and cosine similarity computations, we demonstrated that encrypted systems can achieve high accuracy, with a top-10 match rate of 96.68% and negligible differences from cleartext. The system efficiently handled the biometric comparison of 10,000 individuals while keeping ciphertext sizes under 11 MB.

However, the project also highlighted the computational overhead (8622.54 seconds for encrypted similarity). Future improvements should focus on:

- **FHE Parameter Optimization:** Exploring alternative CKKS configurations.
- **Parallelization/GPU Acceleration:** Reducing runtime for larger datasets.
- **Alternative Encryption Schemes:** Investigating more specialized FHE-based or hybrid approaches.
- **Real-World Deployment:** Testing on broader biometric databases and analyzing robustness under diverse conditions (noise, occlusions, etc.).

Overall, we provide a strong foundation for privacy-preserving biometric applications. Continued research and development are essential for addressing performance bottlenecks and ensuring practical real-world adoption.

# References

[1] J. Bringer, H. Chabanne, and ... An overview of secure multiparty computation in biometric identification. In *Proc. of BTAS*, 2013.

[2] K. Pradel and J. Mitchell. A privacy-preserving biometric authentication protocol with fully homomorphic encryption. arXiv preprint arXiv:2111.12372, 2021.

[3] R. Sperling and ... Heft: Homomorphically encrypted fusion of biometric templates. arXiv preprint arXiv:2208.07241, 2022.

[4] X. Wu and ... Accelerating privacy-preserving biometric identification with tensor triples. Cryptology ePrint Archive, Report 2023/1863, 2023.

# A Optimization Journey

**Optimization Journey: Lessons Learned from Improving Homomorphic Cosine Similarity**

Throughout this project, we explored multiple optimization strategies to enhance performance and accuracy for homomorphic cosine similarity computation.

1. **Initial Naive Approach:** No SIMD packing, each embedding encrypted separately. Accurate but very slow ($\sim$ 1.5 min for 50 test samples).

2. **SIMD Packing:** Dramatically sped up computations but caused severe accuracy drops (below 10%) due to dimension and precision mismatches.

3. **Chunked SIMD + PCA:** Reduced dimensionality to 32/64 but further harmed accuracy ($<$ 30% top-10 match).

4. **Parallelized Homomorphic Computation:** Reduced runtime via multithreading, but synchronization overhead and complexity limited gains.

5. **Final Approach (No Packing + Multithreading + Batch Decryption):** Preserved accuracy ($\sim$ 91.6% top-10) with moderate speed, making it the most balanced approach.

**Key Takeaways:**

- Precision is crucial for sensitive biometric tasks; large approximations degrade accuracy.

- Batching and multithreading provide modest speed-ups without sacrificing accuracy.

- Future work could explore specialized hardware (e.g., GPUs/TPUs) or alternative FHE schemes for further acceleration.

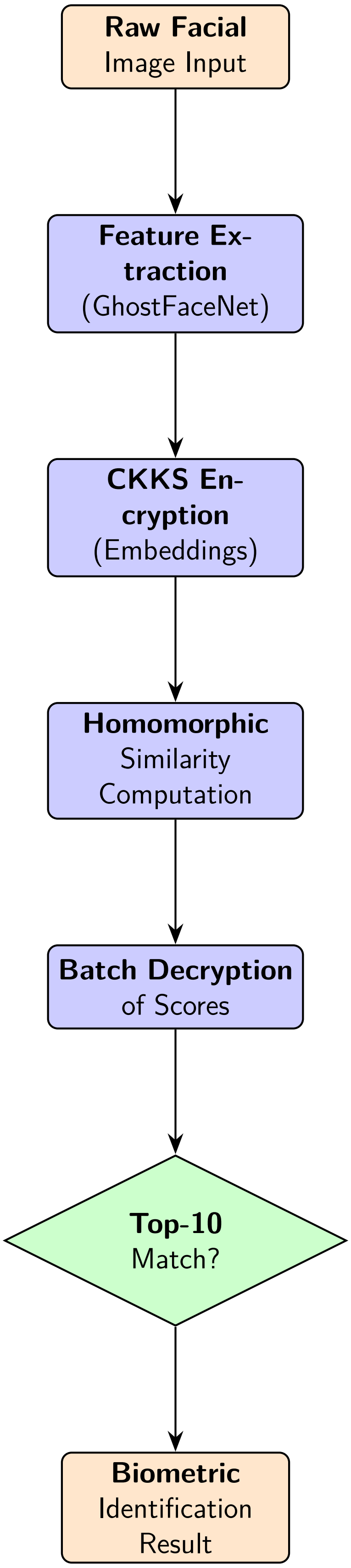
**Raw Facial** Image Input
**Feature Extraction** (GhostFaceNet)
**CKKS Encryption** (Embeddings)
**Homomorphic** Similarity Computation
**Batch Decryption** of Scores
**Top-10** Match?
**Biometric** Identification Result

Figure 1: A more stylized system flow diagram for privacy-preserving biometric identification.