DeepAudit: An Integrity-Aware Deep Facial Recognition System Using Watermarked Embeddings

Abhinaya V, Divya B, Harikrishnan P, Indresan V, Thenmozhi
Department of Computer Science and Engineering KGiSL Institute of Technology, Coimbatore, 641035, India

Abstract- Face recognition systems based on deep learning have achieved remarkable performance and are widely deployed in applications such as access control, surveillance, and attendance management. While existing research primarily focuses on improving recognition accuracy and robustness, relatively little attention has been given to the integrity and security of the training data used by these systems. Since deep learning models are highly dependent on training data, any unauthorized modification or data poisoning can silently degrade system reliability and compromise trust.

This paper presents *DeepAudit*, an integrity-aware deep facial recognition framework that emphasizes securing stored facial embeddings against tampering and data poisoning attacks. Instead of altering the recognition model or training process, the proposed system embeds imperceptible watermarks directly into facial embeddings during the enrollment phase. These watermarks enable post-deployment integrity verification and dataset auditing without affecting real-time recognition performance.

DeepAudit operates in two phases: an enrollment phase, where facial images are processed using a convolutional neural network to generate embeddings that are subsequently watermarked and securely stored, and a recognition phase, where live facial embeddings are matched against stored representations using similarity metrics. The watermark remains passive during recognition and is activated only for integrity verification and audit purposes.

Experimental analysis demonstrates that embedding-level watermarking introduces negligible impact on similarity scores and recognition accuracy while enabling effective detection of unauthorized modifications. The proposed framework highlights the importance of integrating data integrity mechanisms into biometric systems and provides a practical, scalable solution for building trustworthy face recognition pipelines.

I INTRODUCTION

Face recognition has emerged as one of the most prominent and widely deployed biometric technologies in modern digital systems. It plays a critical role in applications such as physical and logical access control, surveillance systems, identity verification, border security, and academic attendance management. The non-intrusive nature of face recognition, combined with its ability to operate without direct user interaction, makes it particularly attractive compared to other biometric modalities such as fingerprints or iris scans.

Recent advancements in deep learning have significantly enhanced the accuracy, robustness, and scalability of face recognition systems. Deep neural networks are capable of learning complex facial representations that are resilient to variations in illumination, pose, facial expression, and aging. As a result, deep learning–based face recognition systems have transitioned from controlled laboratory environments to real-world deployments.

Despite these advancements, security and integrity concerns related to training data remain largely underexplored. Most face recognition pipelines assume that training datasets and stored facial embeddings are trustworthy and remain unaltered after deployment. However, in practical environments where biometric databases are continuously updated, shared, or maintained by multiple stakeholders, this assumption no longer holds. Since machine learning models are highly dependent on the quality and authenticity of their training data, any compromise in this data can directly affect system reliability, accuracy, and trustworthiness.

This section presents a comprehensive overview of face recognition systems, the evolution of deep learning in biometric authentication, the importance of data integrity in machine learning, and the security challenges faced by biometric systems. It further introduces the motivation behind the proposed DeepAudit framework and outlines its primary contributions.

A. Background of Face Recognition Systems

Face recognition systems aim to identify or verify an individual based on distinctive facial characteristics extracted from images or video frames. Early face recognition approaches relied on handcrafted features and statistical models. Techniques such as Eigenfaces and Fisherfaces represented facial images in lower-dimensional subspaces, while texture-based methods like Local Binary Patterns focused on capturing local facial texture information.

Although these traditional methods demonstrated reasonable performance under constrained environments, they struggled in unconstrained real-world conditions involving changes in lighting, facial orientation, occlusions, and image quality. Moreover, handcrafted features lacked the ability to generalize across large and diverse datasets.

Modern face recognition systems have shifted toward learning-based approaches that automatically extract discriminative facial features from data. These systems represent faces as numerical vectors known as embeddings, which capture high-level semantic information about facial identity. Embedding-based recognition enables efficient similarity computation and supports scalable identity matching across large databases, making it suitable for practical deployments.

B. Evolution of Deep Learning in Biometric Authentication

The introduction of deep learning, particularly convolutional neural networks (CNNs), has revolutionized biometric authentication systems. CNNs are designed to automatically learn hierarchical feature representations from raw pixel data, eliminating the need for manual feature engineering. This capability has led to significant improvements in face recognition accuracy and robustness.

Deep embedding models map facial images into high-dimensional feature spaces where embeddings belonging to the same individual are clustered closely, while those of different individuals are well separated. Loss functions such as contrastive loss and triplet loss have further enhanced discriminative learning. These advances have enabled face recognition systems to achieve near-human or even superhuman performance on benchmark datasets.

As a result, deep learning–based biometric systems are increasingly deployed in critical applications, including security-sensitive and large-scale environments. However, their reliance on extensive training data introduces new security risks, particularly related to data manipulation, poisoning, and unauthorized modifications.

C. Importance of Data Integrity in Machine Learning

Data integrity is a fundamental requirement for trustworthy machine learning systems. Training data serves as the foundation upon which models learn patterns and make decisions. If training data is compromised, corrupted, or maliciously altered, the resulting model behavior may become unpredictable or biased.

In biometric systems, compromised training data can lead to severe consequences, including incorrect identity recognition, increased false acceptance rates, and unauthorized access. Unlike performance degradation caused by environmental factors, data integrity violations may remain undetected for long periods, silently undermining system reliability.

Ensuring the authenticity and integrity of stored training data and derived embeddings is therefore essential. Effective integrity verification mechanisms can detect unauthorized modifications and provide auditability, thereby enhancing trust in machine learning-driven biometric systems.

D. Security Challenges in Biometric Systems

Biometric systems face a wide range of security challenges throughout their lifecycle. Data poisoning attacks involve injecting malicious or misleading samples into training datasets with the intention of degrading recognition performance or causing targeted misclassification. Such attacks can be particularly damaging when executed subtly, as they may evade traditional detection mechanisms.

Insider threats pose an equally significant risk, as authorized personnel may intentionally or unintentionally modify stored biometric data. Additionally, biometric databases may be vulnerable to unauthorized access, data replacement, or tampering due to inadequate security controls.

Traditional face recognition systems primarily focus on improving recognition accuracy and efficiency, often overlooking the security of stored embeddings. The assumption that training data remains static and trustworthy after deployment is unrealistic in dynamic, real-world environments where databases are frequently updated and maintained.

E. Motivation Behind DeepAudit

The motivation behind DeepAudit stems from the need to move beyond accuracy-centric face recognition systems toward integrity-aware biometric frameworks. While significant research has focused on improving recognition performance, comparatively little attention has been given to auditing and protecting the training data itself.

DeepAudit addresses this gap by introducing an embedding-level watermarking mechanism that secures stored facial embeddings against tampering and unauthorized modification. By embedding imperceptible watermarks into facial representations, the system enables post-deployment integrity verification without interfering with real-time recognition.

F. Contributions of the Proposed System

The key contributions of the proposed DeepAudit framework are summarized as follows:

- Design of an integrity-aware face recognition system based on deep facial embeddings.
- Introduction of embedding-level watermarking to protect training data from tampering and poisoning.
- Development of a passive auditing mechanism that does not affect recognition accuracy.
- Analysis of the impact of watermarking on similarity metrics and recognition performance.
- Emphasis on integrating data security as a core component of biometric system design.

II Literature Review

Face recognition and biometric authentication have been extensively studied in the literature, with research efforts primarily focused on improving recognition accuracy, robustness, and scalability. Early studies explored handcrafted feature extraction techniques, which laid the foundation for subsequent developments but exhibited limited performance under unconstrained conditions.

The emergence of deep learning marked a significant paradigm shift in face recognition research. CNN-based models demonstrated remarkable improvements by learning discriminative facial features directly from large-scale datasets. Numerous studies have proposed deep embedding frameworks that achieve state-of-the-art performance on standard benchmarks.

Despite these advancements, most existing research assumes that training data and stored embeddings remain clean and trustworthy. Security concerns related to training data integrity, particularly in post-deployment scenarios, are often neglected. This oversight poses serious risks, as compromised training data can silently degrade system performance.

Recent studies have highlighted the vulnerability of machine learning systems to data poisoning attacks. In biometric contexts, poisoning attacks may involve injecting mislabeled facial samples or modifying stored embeddings to influence recognition outcomes. Existing defenses largely focus on anomaly detection during training, which may not be effective against subtle attacks or insider-driven modifications.

Digital watermarking has been widely used in multimedia security to protect images, audio, and video content. Some recent works have extended watermarking concepts to neural networks and model ownership verification. However, the application of watermarking directly to biometric embeddings for integrity verification remains relatively unexplored.

The reviewed literature reveals a clear research gap in integrating data integrity and auditability mechanisms into face recognition pipelines. Most existing systems prioritize recognition performance while overlooking the security of stored biometric representations. The proposed DeepAudit framework addresses this gap by combining deep facial recognition with embedding-level watermarking, offering a novel and practical approach to secure and auditable biometric authentication.

A. Overview of Face Recognition Techniques

Face recognition techniques have evolved significantly over the past few decades, driven by advances in computer vision, pattern recognition, and machine learning. Broadly, these techniques can be categorized into traditional feature-based methods and modern deep learning–based approaches. Each category reflects a distinct philosophy in how facial information is represented, learned, and compared for identity recognition.

B. Traditional Methods

Traditional face recognition methods primarily relied on handcrafted feature extraction combined with statistical learning techniques. These approaches aimed to manually design features that capture distinctive facial characteristics while reducing dimensionality and noise. One of the earliest and most influential methods, Eigenfaces, utilized Principal Component Analysis (PCA) to represent facial images in a lower-dimensional subspace. By projecting faces onto a set of orthogonal basis vectors, Eigenfaces captured the dominant global variations across facial images.

Fisherfaces extended this concept by incorporating Linear Discriminant Analysis (LDA), which emphasized maximizing interclass separability while minimizing intra-class variance. This made Fisherfaces more robust to lighting variations compared to Eigenfaces. However, both approaches relied heavily on global facial representations, making them sensitive to misalignment, pose variations, and occlusions.

To address these limitations, texture-based methods gained prominence. Local Binary Patterns (LBP) encoded local texture information by comparing pixel intensities within small neighborhoods, enabling better resilience to illumination changes. Variants of LBP were widely adopted due to their computational simplicity and reasonable performance in constrained environments. Similarly, Gabor filters were employed to capture facial features at multiple scales and orientations, mimicking human visual perception. Histogram of Oriented Gradients (HOG) further contributed by encoding gradient direction distributions, particularly useful for capturing facial contours and edge structures.

Despite these advancements, traditional methods exhibited several inherent limitations. Their reliance on handcrafted features restricted adaptability across diverse datasets. Performance degraded significantly in unconstrained environments involving variations in pose, expression, aging, and partial occlusions. Additionally, these methods struggled to scale effectively to large datasets, which are common in modern biometric applications. These shortcomings ultimately motivated the transition toward data-driven learning approaches.

C. Deep Learning–Based Methods

Deep learning–based face recognition methods have largely superseded traditional techniques due to their superior representational power and robustness. Convolutional Neural Networks (CNNs) introduced an end-to-end learning paradigm, enabling models to automatically learn hierarchical facial features directly from raw pixel data. Lower layers capture basic visual patterns such as edges and textures, while deeper layers encode higher-level semantic features related to facial identity.

Unlike handcrafted methods, deep learning models adapt their feature representations based on data, allowing them to generalize effectively across varying conditions. Deep face recognition systems typically transform facial images into compact numerical representations known as embeddings. These embeddings are learned such that samples belonging to the same individual are clustered closely in the embedding space, while those from different individuals are well separated.

This embedding-based paradigm supports scalable identity matching, as similarity computation can be efficiently performed using distance metrics such as cosine similarity or Euclidean distance. Deep learning–based methods have consistently achieved state-of-the-art performance on benchmark datasets and have been widely deployed in real-world systems, including mobile authentication, surveillance, and attendance monitoring.

However, despite their success, deep learning–based systems exhibit a critical dependency on training data quality and integrity. Since embeddings are learned directly from data, any compromise in training samples or stored embeddings can significantly affect recognition reliability, making these systems vulnerable to data-centric attacks.

D. CNN-Based Facial Embedding Models

CNN-based facial embedding models constitute the core of modern face recognition systems. These models consist of stacked convolutional, pooling, and normalization layers designed to extract progressively abstract facial features. The final layers generate fixed-length embedding vectors that serve as unique representations of facial identity.

To enhance discriminative capability, various embedding learning strategies have been proposed. Contrastive loss enforces distance constraints between pairs of samples, while triplet loss optimizes relative distances among anchor, positive, and negative samples. More recent angular margin–based loss functions introduce explicit geometric constraints in the embedding space, leading to improved class separation and robustness.

These embedding models enable efficient one-to-many and many-to-many identity matching, making them suitable for large-scale biometric databases. However, once embeddings are generated during enrollment, they are typically stored and reused for recognition without additional verification. The assumption that these stored embeddings remain static and trustworthy introduces a critical security vulnerability.

Embedding databases may be subject to unauthorized modification, replacement, or poisoning, especially in systems with frequent updates or multiple administrators. The absence of integrity validation mechanisms makes CNN-based embedding models particularly susceptible to post-deployment attacks.

E. Training Data Vulnerabilities in Machine Learning

Training data plays a foundational role in machine learning systems, as model behavior is inherently shaped by the data used during training. Any compromise in training data quality, authenticity, or integrity can result in degraded performance, biased predictions, or unintended system behavior.

Machine learning systems are especially vulnerable to data-centric attacks because training datasets are often aggregated from diverse and distributed sources. In biometric applications, new identities may be enrolled continuously, increasing the risk of unauthorized data injection or modification. Furthermore, stored embeddings derived from training data are often treated as static assets, despite being critical to system operation.

Unlike model-level attacks, training data manipulation may not produce immediate or observable anomalies. As a result, compromised models may continue to operate while silently producing incorrect recognition outcomes. These vulnerabilities highlight the importance of implementing mechanisms that ensure training data integrity throughout the entire system lifecycle, including post-deployment phases.

F. Data Poisoning Attacks in Biometric Systems

Data poisoning attacks represent a significant threat to biometric systems. Such attacks involve injecting malicious samples or modifying existing data to influence model behavior. In face recognition systems, attackers may target raw facial images, identity labels, or stored facial embeddings.

One common poisoning strategy involves inserting mislabeled facial images to cause systematic misclassification. More advanced attacks aim to subtly alter decision boundaries while preserving overall system accuracy, making detection challenging. These attacks can be particularly damaging in security-sensitive applications such as access control and surveillance.

Insider threats further exacerbate the problem, as authorized personnel may intentionally or unintentionally manipulate biometric data. Traditional defense mechanisms primarily focus on detecting anomalous samples during training, which may be ineffective against subtle or insider-driven poisoning attacks.

These challenges underscore the need for post-training integrity verification mechanisms that can detect unauthorized modifications to stored biometric representations, rather than relying solely on training-time defenses.

G. Existing Security Mechanisms in Face Recognition

Existing security mechanisms in face recognition systems largely focus on protecting communication channels, encrypting biometric databases, and enforcing access control policies. Encryption ensures confidentiality during storage and transmission but does not guarantee integrity once data is accessed or decrypted.

Some research efforts propose robust training techniques and anomaly detection methods to mitigate poisoning attacks. However, these approaches often require retraining models or maintaining trusted reference datasets, which may not be feasible in real-world deployments with limited resources.

Importantly, most current face recognition systems lack built-in mechanisms to audit stored embeddings or verify their authenticity. As a result, unauthorized modifications may remain undetected for extended periods, leading to long-term degradation in system reliability and security.

H. Digital Watermarking Techniques in Machine Learning

Digital watermarking has been widely used in multimedia security to protect images, audio, and video content from unauthorized modification and redistribution. Watermarking techniques embed imperceptible information into digital content, enabling ownership verification, tamper detection, and integrity checking.

Recent studies have explored watermarking in machine learning contexts, including embedding watermarks into neural network weights or outputs to protect intellectual property. These techniques primarily focus on model ownership verification rather than data integrity.

Embedding watermarks directly into biometric embeddings presents a promising yet underexplored direction. Embedding-level watermarking enables integrity verification of stored data without modifying model architecture or affecting recognition performance. This makes it particularly suitable for biometric systems where accuracy and efficiency are critical.

I. Research Gaps Identified

The literature review reveals several critical research gaps. First, existing face recognition systems prioritize accuracy, scalability, and efficiency while largely neglecting training data integrity. Second, most defenses against data poisoning focus on training-time detection and fail to address post-deployment tampering. Third, existing security mechanisms do not provide embedding-level auditability or integrity verification.

These gaps highlight the need for a face recognition framework that integrates integrity verification directly into the biometric pipeline. The proposed DeepAudit system addresses these limitations by introducing watermarking at the embedding level, enabling secure storage, auditing, and verification of facial embeddings without compromising recognition accuracy. This approach provides a novel and practical foundation for building trustworthy and integrity-aware biometric systems.

Table 1: Comparison of Existing Face Recognition Methods

Author	**Method**	**Accuracy**	**Security**
Taigman et al.	DeepFace	97.3%	No
Schroff et al.	FaceNet	99.6%	No
Deng et al.	ArcFace	99.8%	No
DeepAudit (Ours)	**CNN + WM**	**High**	**Yes**

III Problem Definition and Threat Analysis

Deep learning–based face recognition systems have achieved remarkable success in terms of accuracy and scalability, leading to their widespread adoption in security-sensitive applications such as access control, surveillance, identity verification, and attendance management. Despite these advancements, security concerns related to the integrity of training data and stored facial embeddings remain largely unaddressed. Most existing systems emphasize recognition performance and computational efficiency, implicitly assuming that training data is trustworthy and remains unchanged after enrollment.

In real-world deployments, this assumption is frequently violated. Face recognition systems often operate in dynamic environments where data is continuously updated, multiple administrators have access to biometric databases, and systems are exposed to both external and internal threats. Under such conditions, training data and stored embeddings become vulnerable to unauthorized modification, replacement, or injection. This section defines the core problem addressed in this work and presents a detailed analysis of the threat landscape associated with compromised training data in face recognition systems.

A. Problem Definition

The primary problem addressed in this work is the absence of effective mechanisms to ensure the integrity and authenticity of training data in deep learning–based face recognition systems. Once facial images are processed during enrollment and converted into embeddings, these representations are typically stored in a database and reused for recognition without any form of integrity verification. Any unauthorized modification to these embeddings can directly influence recognition decisions.

Unlike traditional software systems, where integrity checks and validation mechanisms are well established, machine learning systems lack inherent capabilities to verify the correctness or authenticity of their training data. As a result, compromised data may not trigger explicit errors or alerts. Instead, it may cause gradual degradation in performance or enable targeted misclassification of specific identities.

In biometric systems, such silent failures pose severe security risks. Unauthorized access, identity spoofing, and denial of service are possible outcomes of compromised training data. The challenge lies in designing a system that can verify the integrity of stored facial embeddings without retraining the recognition model, modifying its architecture, or degrading recognition accuracy. Existing solutions largely overlook post-deployment data integrity, creating a critical gap that the proposed DeepAudit framework seeks to address.

B. Assumptions in Traditional Face Recognition Systems

Traditional face recognition systems are built upon several implicit assumptions regarding training data, system operation, and trust boundaries. One key assumption is that enrollment data is collected from trusted sources and remains unaltered after being stored in the database. Another assumption is that system administrators and internal users will not intentionally or unintentionally modify stored biometric data.

While these assumptions may hold in controlled laboratory environments or small-scale deployments, they are unrealistic in real-world scenarios. In practice, face recognition systems are often deployed for extended periods, updated incrementally, and maintained by multiple stakeholders. Integration with external data sources and third-party services further increases exposure to potential manipulation.

Additionally, most systems assume that any compromise in training data will result in immediately observable performance degradation. However, subtle modifications to facial embeddings may not significantly affect overall accuracy metrics, allowing attacks to persist undetected. These flawed assumptions highlight the necessity of incorporating explicit integrity verification mechanisms into face recognition pipelines.

C. Data Poisoning Threat Model

Data poisoning attacks constitute a major threat to machine learning systems. In face recognition applications, data poisoning may occur at various stages, including facial image acquisition, embedding generation, and database storage. Attackers may inject malicious samples, alter identity labels, or directly manipulate stored embeddings to influence recognition outcomes.

A common poisoning strategy involves inserting mislabeled facial images into the training dataset, causing the system to associate incorrect identities. More sophisticated attacks aim to subtly modify embeddings in a manner that preserves overall system accuracy while enabling targeted misclassification of specific individuals. Such attacks are particularly dangerous because they are difficult to detect using traditional anomaly detection techniques.

The threat model considered in this work assumes that attackers may gain partial or indirect access to the embedding database but do not have control over the recognition model itself. This assumption reflects realistic attack scenarios in which databases are exposed through misconfiguration, insider access, or software vulnerabilities. Under this model, ensuring the integrity of stored embeddings becomes a critical defense mechanism. DeepAudit addresses this threat by embedding imperceptible watermarks into facial representations, enabling detection of unauthorized modifications.

D. Insider Threats and Unauthorized Data Modification

Insider threats pose a particularly severe risk to biometric systems due to the privileged access enjoyed by authorized users. Administrators, operators, or maintenance personnel may intentionally or unintentionally modify stored biometric data as a result of misconfiguration, negligence, or malicious intent. Since insiders operate within legitimate access boundaries, their actions are often difficult to distinguish from normal system behavior.

Unauthorized data modification may include replacing embeddings, altering identity-to-embedding mappings, or deleting records. Such actions can compromise recognition reliability and undermine system trust. Traditional security mechanisms primarily focus on protecting against external attackers and often fail to address insider-driven threats.

Most existing face recognition systems do not provide mechanisms to audit stored embeddings or verify their authenticity over time. Consequently, insider-driven modifications can persist undetected for extended periods, potentially causing long-term security breaches. DeepAudit introduces embedding-level auditability, enabling verification of stored data regardless of whether the modification originates from an external attacker or an insider.

E. Impact of Compromised Training Data

Compromised training data can have far-reaching consequences on face recognition system performance, reliability, and trustworthiness. Even minor alterations to stored embeddings may increase false acceptance or false rejection rates, particularly for targeted identities. In security-critical applications, such failures can result in unauthorized access, impersonation, or denial of legitimate services.

Beyond immediate recognition errors, compromised data undermines user confidence in biometric systems. Unlike passwords or tokens, biometric identifiers cannot be easily replaced, making data integrity violations especially damaging. Additionally, compromised systems may exhibit biased behavior, disproportionately affecting certain individuals or groups, thereby raising ethical and legal concerns.

These impacts demonstrate that protecting training data is as crucial as improving recognition accuracy. A system that performs well under ideal conditions but fails when training data integrity is compromised cannot be considered secure or trustworthy.

F. Need for Auditable Face Recognition Systems

The analysis of existing threats highlights the urgent need for face recognition systems that support auditability and integrity verification. Auditable systems provide mechanisms to verify that stored biometric data remains authentic and unaltered throughout the system lifecycle. Such capabilities are essential for detecting tampering, investigating security incidents, and maintaining long-term trust.

An effective auditable face recognition system should enable post-deployment integrity checks without disrupting real-time recognition or imposing significant computational overhead. It should also operate independently of the recognition model, avoiding retraining or architectural changes. DeepAudit fulfills these requirements by embedding watermarks into facial embeddings, enabling passive auditing and reliable integrity verification while preserving recognition performance.

IV System Objectives and Design Goals

The design of DeepAudit is guided by the need to address the security and reliability challenges identified in existing face recognition systems while preserving their operational efficiency and accuracy. Unlike traditional approaches that treat security as an auxiliary concern, DeepAudit integrates integrity verification as a core component of the biometric pipeline. This section outlines the primary objectives of the proposed system and the design goals that shape its architecture, functionality, and deployment strategy.

A. Primary Objectives

The primary objective of DeepAudit is to ensure the integrity and authenticity of stored facial embeddings used in deep learning–based face recognition systems. The framework is designed to protect training data against unauthorized modification, replacement, and data poisoning attacks that may occur during or after system deployment. By focusing on embedding-level protection, DeepAudit addresses vulnerabilities that are often overlooked in conventional recognition pipelines.

Another key objective is to introduce auditability into the face recognition process. The system aims to provide mechanisms that enable post-deployment verification of stored biometric data without disrupting real-time recognition operations. This capability is essential for detecting tampering, conducting forensic analysis, and maintaining long-term trust in biometric systems.

DeepAudit also seeks to operate independently of the underlying recognition model. The proposed framework does not require retraining, fine-tuning, or architectural modification of the face recognition network. This design choice ensures compatibility with a wide range of existing CNN-based embedding models and facilitates seamless integration into deployed systems.

B. Security-Oriented Design Goals

Security is a central design consideration in DeepAudit. The system adopts a defense-in-depth approach by embedding integrity verification directly into facial embeddings rather than relying solely on external security controls such as encryption or access management. This approach ensures that data integrity can be verified even after decryption or internal access.

Embedding-level watermarking is employed as the primary security mechanism. The watermark is designed to be imperceptible to the recognition process, ensuring that it does not influence similarity computation or decision-making. At the same time, the watermark must be robust enough to detect unauthorized modifications to embeddings, whether caused by external attackers or insider threats.

The design also assumes minimal trust in storage environments and administrative users. By embedding integrity information within the data itself, DeepAudit enables verification regardless of where or how the data is stored. This assumption aligns with realistic deployment scenarios in which biometric databases may be distributed, shared, or exposed to partial compromise.

Additionally, the system is designed to support passive auditing. Integrity checks are performed only when required, such as during periodic audits or incident investigations, rather than during every recognition operation. This minimizes performance overhead while maintaining strong security guarantees.

C. Accuracy and Performance Constraints

A critical design goal of DeepAudit is to preserve the accuracy and performance of the face recognition system. Since biometric applications often operate in real-time environments, any security mechanism that introduces significant computational overhead or degrades recognition accuracy is impractical.

To address this constraint, the watermarking process is carefully designed to introduce minimal perturbation to facial embeddings. The magnitude and structure of the watermark are selected such that similarity scores computed during recognition remain largely unchanged. As a result, threshold-based matching decisions are preserved, and recognition accuracy is not adversely affected.

The system avoids additional processing during the recognition phase. Live facial embeddings are compared directly with stored, watermarked embeddings using standard distance metrics. Integrity verification is decoupled from recognition and performed only during audit operations. This separation ensures that real-time performance requirements are met.

Furthermore, DeepAudit is designed to be computationally lightweight. The watermark embedding and extraction processes operate on fixed-length vectors and do not involve complex transformations or iterative optimization. This design choice ensures that the system can be deployed on resource-constrained platforms without compromising efficiency.

D. Scalability and Practical Deployment Goals

Scalability and practicality are essential considerations for real-world biometric systems. DeepAudit is designed to support large-scale deployments involving thousands or millions of enrolled identities. The framework accommodates incremental enrollment, allowing new users to be added without affecting existing embeddings or requiring system downtime.

The modular design of DeepAudit enables easy integration with existing face recognition pipelines. Each component, including feature extraction, watermark embedding, storage, recognition, and auditing, operates independently. This modularity facilitates system maintenance, upgrades, and customization for different application scenarios.

From a deployment perspective, DeepAudit does not impose strict requirements on storage infrastructure or network architecture. Watermarked embeddings can be stored in conventional databases, and integrity verification can be performed offline or on-demand. This flexibility makes the system suitable for a wide range of environments, including academic institutions, enterprises, and public infrastructure.

Finally, the design goals emphasize long-term maintainability and adaptability. DeepAudit can be extended to support additional biometric modalities or integrated with complementary security mechanisms such as multi-factor authentication or secure logging. By balancing security, performance, and scalability, the proposed framework provides a practical foundation for deploying trustworthy face recognition systems in real-world settings.

V Overall System Architecture

The architecture of DeepAudit is designed to integrate integrity verification seamlessly into a deep learning–based face recognition pipeline. Unlike conventional systems that focus solely on recognition accuracy, the proposed architecture emphasizes secure handling, storage, and auditing of facial embeddings throughout the system lifecycle. The architecture adopts a modular and layered approach to ensure scalability, maintainability, and compatibility with existing face recognition frameworks.

This section presents an overview of the DeepAudit architecture, discusses its modular design philosophy, explains data flow across components, and describes the architectures of the enrollment phase, recognition phase, and secure embedding storage.

A. Architectural Overview of DeepAudit

DeepAudit consists of a set of interconnected modules that collectively enable face recognition with embedding-level integrity verification. At a high level, the system processes facial images to generate embeddings using a convolutional neural network, embeds watermarks into these representations during enrollment, and stores them in a secure database. During recognition, live facial embeddings are matched against stored embeddings using similarity measures, while watermark verification remains passive and is invoked only during auditing.

The architecture separates recognition functionality from integrity verification, ensuring that security mechanisms do not interfere with real-time performance. This separation allows DeepAudit to operate transparently within existing recognition systems while adding an additional layer of trust and auditability.

B. Modular Design Philosophy

DeepAudit follows a modular design philosophy in which each system component performs a specific function and interacts with other modules through well-defined interfaces. This design improves system flexibility and allows individual modules to be updated or replaced without affecting the overall system.

Key modules in the DeepAudit architecture include the image acquisition module, face detection module, feature extraction module, watermarking module, secure embedding database, and matching and decision module. By decoupling these components, the system supports independent development, testing, and maintenance.

The modular approach also enhances scalability. As system requirements evolve, additional modules such as multi-factor authentication, logging mechanisms, or external auditing services can be integrated without redesigning the core architecture. This philosophy ensures that DeepAudit remains adaptable to diverse deployment scenarios.

C. Data Flow in DeepAudit

The data flow in DeepAudit follows a structured pipeline that governs how facial data is processed, stored, and verified. During the enrollment phase, facial images are captured, preprocessed, and passed through the feature extraction module to generate embeddings. These embeddings are then processed by the watermarking module, which embeds integrity-related information before storage.

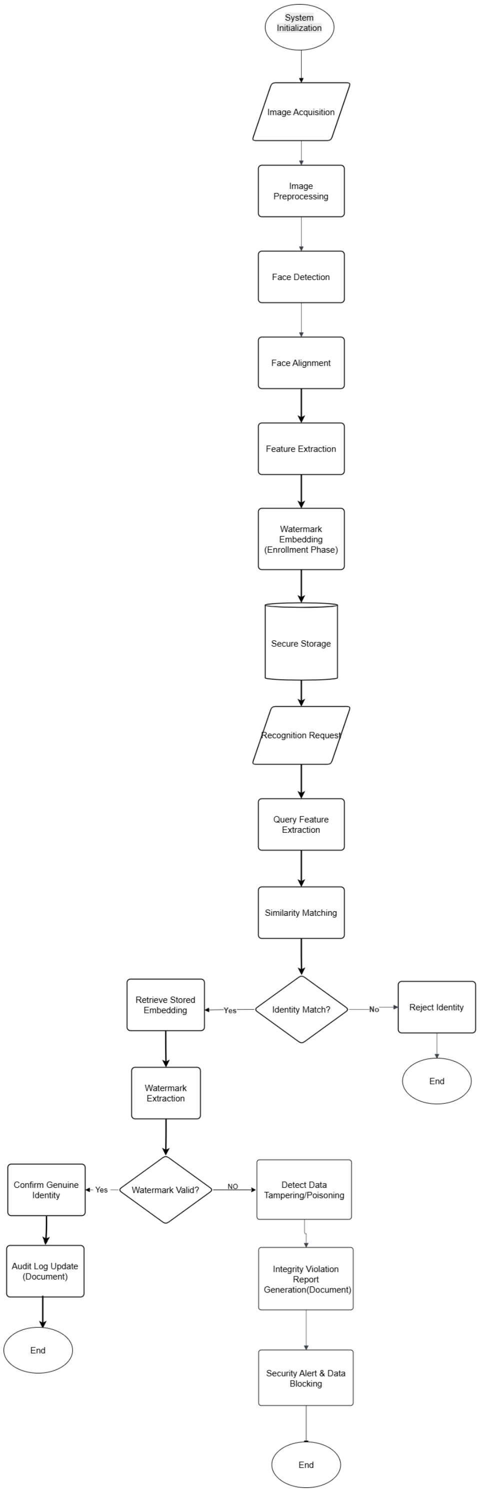

Figure 1: Overall Framework of DeepAudit

The Figure 1 describes a secure face recognition system with an added watermarking mechanism to protect identity data. The process starts with capturing and preprocessing an image, followed by face detection, alignment, and feature extraction. During enrollment, a digital watermark is embedded into the extracted features and stored securely. When a recognition request is made, features from the query image are matched with stored data. If no match is found, the identity is rejected. If a match is found, the watermark is extracted and checked. A valid watermark confirms a genuine identity and updates the audit log, while an invalid watermark indicates possible tampering, leading to security alerts and data blocking.

During the recognition phase, live facial images follow a similar preprocessing and feature extraction pipeline. However, the resulting embeddings are directly compared with stored embeddings for identity matching. The watermark embedded in stored embeddings remains inactive during this process and does not influence similarity computation.

Audit operations form a separate data flow path. When an audit is initiated, stored embeddings are retrieved from the database and passed to the watermark verification module. Any discrepancy between the expected and extracted watermark indicates potential tampering or unauthorized modification.

D. Enrollment Phase Architecture

The enrollment phase architecture is responsible for securely registering new identities into the system. Facial images are first acquired through a camera or image input device and passed to the face detection module, which locates and extracts the facial region. Preprocessing operations such as alignment and normalization are applied to ensure consistency.

The processed facial image is then forwarded to the CNN-based feature extraction module, which generates a fixed-length embedding representing the individual's facial identity. This embedding serves as the primary biometric representation used for recognition.

Before storage, the embedding is processed by the watermarking module. The watermark encodes integrity-related information, such as identity metadata or cryptographic signatures, and is embedded in a manner that minimizes distortion. The watermarked embedding is then stored in the secure embedding database. This architecture ensures that all enrolled data is protected from the moment it enters the system.

E. Recognition and Verification Phase Architecture

The recognition and verification phase architecture enables real-time identification or authentication of individuals. Live facial images are captured and processed using the same face detection and feature extraction pipeline employed during enrollment. This consistency ensures that embeddings are comparable across phases.

The live embedding is compared against stored embeddings using similarity metrics such as cosine similarity or Euclidean distance. A predefined threshold determines whether a match is accepted or rejected. Importantly, the watermark embedded in stored embeddings does not participate in the matching process, ensuring that recognition accuracy and performance remain unaffected.

Integrity verification is not performed during routine recognition to avoid additional overhead. Instead, watermark verification is triggered only during audit operations or when suspicious activity is detected. This design preserves real-time performance while maintaining strong security guarantees.

F. Secure Embedding Storage Architecture

The secure embedding storage architecture is a critical component of DeepAudit. Watermarked embeddings are stored in a database that may reside on local servers or cloud infrastructure. While traditional security measures such as encryption and access control may be employed, DeepAudit does not rely solely on external protections.

By embedding integrity information directly into the data, the system ensures that tampering can be detected even if storage-level security is compromised. Stored embeddings can be periodically audited or verified on demand to confirm their authenticity.

The storage architecture supports scalability and incremental updates, allowing new embeddings to be added without affecting existing records. This design enables long-term deployment of DeepAudit in dynamic environments while maintaining data integrity and auditability.

VI Face Detection and Feature Extraction

Face detection and feature extraction form the foundational stages of any face recognition system. The reliability and accuracy of the overall recognition pipeline are highly dependent on the quality of facial region localization and the discriminative power of extracted features. In the DeepAudit framework, these stages are designed to ensure consistent and robust facial representation while maintaining compatibility with the proposed integrity verification mechanisms.

This section describes the processes involved in image acquisition, face detection, facial alignment and normalization, CNN-based feature extraction, and facial embedding representation. It also highlights the advantages of embedding-based recognition in the context of secure biometric systems.

A. Image Acquisition and Preprocessing

Image acquisition refers to the process of capturing facial images from input sources such as cameras, video streams, or stored image files. In real-world deployments, facial images may be acquired under varying lighting conditions, backgrounds, and camera resolutions. These variations can significantly affect recognition performance if not handled appropriately.

To mitigate such effects, preprocessing operations are applied to raw images before further processing. Preprocessing typically includes resizing images to a fixed resolution, converting color images to standardized formats, and reducing noise through filtering techniques. These steps ensure uniformity across input samples and improve the stability of subsequent detection and feature extraction stages.

In DeepAudit, preprocessing is designed to be lightweight and consistent across enrollment and recognition phases. This consistency ensures that embeddings generated during enrollment and recognition remain comparable, thereby preserving recognition accuracy.

B. Face Detection Techniques Used

Face detection is the process of locating facial regions within an image or video frame. Accurate face detection is critical, as errors at this stage propagate through the recognition pipeline. Traditional face detection methods relied on handcrafted features and sliding-window classifiers, which often struggled under unconstrained conditions.

Modern face recognition systems employ learning-based face detection techniques that offer improved robustness and accuracy. These detectors can handle variations in pose, illumination, and partial occlusions more effectively than traditional methods. The output of the face detection module is a bounding box that encloses the detected facial region.

In the DeepAudit framework, face detection is treated as an independent module, allowing flexibility in selecting or upgrading detection algorithms without affecting other system components. This modularity supports adaptation to different deployment environments and hardware capabilities.

C. Facial Alignment and Normalization

Once a face is detected, facial alignment and normalization are performed to reduce variations caused by pose and orientation. Facial alignment involves locating key facial landmarks such as the eyes, nose, and mouth and using them to align the face to a canonical orientation.

Normalization techniques include geometric transformations such as rotation, scaling, and cropping to ensure that facial features are consistently positioned across images. Illumination normalization may also be applied to reduce the impact of lighting variations.

These operations improve the consistency of facial representations and enhance the discriminative power of extracted features. In DeepAudit, alignment and normalization are applied uniformly during both enrollment and recognition, ensuring that embeddings remain stable and comparable.

D. CNN-Based Feature Extraction

CNN-based feature extraction is a core component of modern face recognition systems. Convolutional neural networks are capable of learning hierarchical feature representations directly from pixel-level input. Early layers capture low-level patterns such as edges and textures, while deeper layers encode higher-level semantic information related to facial identity.

In the DeepAudit framework, a pre-trained CNN model is used to extract deep facial features from normalized facial images. The network processes each image and outputs a fixed-length feature vector, commonly referred to as a facial embedding. These embeddings capture identity-specific information while being invariant to non-discriminative factors such as lighting and minor pose variations.

Using a pre-trained model allows DeepAudit to leverage state-of-the-art recognition performance without requiring extensive retraining. This approach also ensures compatibility with existing face recognition pipelines and facilitates integration into deployed systems.

E. Facial Embedding Representation

Facial embeddings are numerical vector representations that uniquely encode an individual's facial characteristics. Each embedding resides in a high-dimensional feature space where the distance between embeddings reflects facial similarity. Embeddings belonging to the same identity are expected to cluster closely, while those of different identities are well separated.

This representation enables efficient and scalable matching using distance-based similarity measures. Unlike raw images, embeddings require significantly less storage and are more suitable for large-scale biometric databases.

In DeepAudit, facial embeddings serve as the primary biometric data unit. Integrity protection mechanisms, including watermarking, are applied directly to these embeddings. This design choice allows the system to verify data authenticity without interfering with the feature extraction or recognition process.

F. Advantages of Embedding-Based Recognition

Embedding-based recognition offers several advantages over traditional template-matching approaches. First, embeddings provide compact and discriminative representations that support efficient storage and fast similarity computation. Second, they enable scalable one-to-many matching, making them suitable for large biometric databases.

From a security perspective, embedding-based systems allow integrity verification mechanisms to be applied at the data representation level. In DeepAudit, embedding-level watermarking enables detection of unauthorized modifications without altering recognition accuracy or system performance.

Additionally, embedding-based recognition supports modular system design. Feature extraction, matching, and integrity verification can be treated as independent components, enhancing flexibility and maintainability. These advantages make embedding-based recognition an effective foundation for building secure and trustworthy biometric systems.

VII Training Data Preparation and Management

Training data preparation and management play a crucial role in determining the reliability, accuracy, and security of face recognition systems. Since deep learning models derive their decision-making capability directly from training data, improper data handling or compromised data integrity can significantly degrade system performance. In the DeepAudit framework, training data preparation is treated as a security-sensitive process rather than a one-time preprocessing step.

This section describes the dataset collection strategy, handling of facial variations, embedding generation process, secure storage of training embeddings, and the risks associated with embedding storage in face recognition systems.

A. Dataset Collection Strategy

The dataset collection strategy aims to acquire representative and diverse facial samples for each enrolled individual. Multiple images are collected per subject to capture natural variations in appearance and environmental conditions. This diversity is essential for training robust facial embeddings that generalize well during recognition.

In practical deployments, facial images may be captured using cameras, mobile devices, or existing image repositories. The collection process must ensure that images are obtained in a controlled and ethical manner, with proper consent and compliance with privacy regulations. Poor-quality images or mislabeled samples can introduce noise and bias into the system, negatively impacting recognition accuracy.

DeepAudit assumes a supervised enrollment process in which identity labels are verified at the time of data collection. This assumption reduces the risk of incorrect identity associations during enrollment while allowing the system to focus on protecting data integrity after storage.

B. Handling Variations in Facial Images

Facial images exhibit significant variability due to factors such as pose, illumination, facial expression, aging, occlusions, and camera resolution. Handling these variations is essential to ensure consistent embedding generation and reliable recognition.

To address pose and expression variations, multiple images per individual are captured under different conditions. Illumination variations are mitigated through preprocessing techniques such as normalization and contrast adjustment. Facial alignment further reduces variability by positioning facial landmarks into a canonical form.

By explicitly accounting for natural variations during dataset preparation, DeepAudit reduces the likelihood that recognition errors are caused by environmental factors rather than data integrity issues. This distinction is important when auditing system behavior and diagnosing potential attacks.

C. Embedding Generation Process

Once facial images are collected and preprocessed, they are converted into numerical representations through a CNN-based feature extraction process. Each image is passed through a pretrained convolutional neural network, which outputs a fixed-length embedding vector representing the individual's facial identity.

Multiple embeddings may be generated for each individual to capture intra-class variability. These embeddings collectively form the training dataset used during recognition. Aggregation strategies, such as averaging or selecting representative embeddings, may be employed depending on system requirements.

In the DeepAudit framework, embedding generation is performed prior to watermarking. This separation ensures that the recognition model operates on standard embeddings, while integrity protection mechanisms are applied only at the storage level. This design choice preserves compatibility with existing recognition models and avoids retraining.

D. Secure Storage of Training Embeddings

Secure storage of training embeddings is a critical requirement for face recognition systems. Embeddings are typically stored in databases that support fast retrieval and similarity computation. While conventional security measures such as encryption, authentication, and access control may be applied, these measures alone do not guarantee data integrity.

DeepAudit enhances storage security by embedding integrity-related information directly into the embeddings before storage. Watermarked embeddings carry verification information that enables detection of unauthorized modification, replacement, or deletion. This approach ensures that integrity verification remains possible even if storage-level protections are bypassed.

The storage architecture supports incremental updates, allowing new embeddings to be added without affecting existing records. Periodic audits or on-demand verification can be performed to assess the authenticity of stored embeddings and identify potential tampering.

E. Risks Associated with Embedding Storage

Despite their compactness and efficiency, facial embeddings introduce unique security risks. Since embeddings are numerical representations rather than raw images, they may be mistakenly treated as non-sensitive data. However, compromised embeddings can be exploited to manipulate recognition outcomes or reconstruct sensitive information.

Unauthorized modification of embeddings may lead to targeted misclassification, increased false acceptance rates, or denial of service for legitimate users. Insider threats further exacerbate these risks, as authorized users may have direct access to embedding databases.

Another risk arises from the lack of visibility into embedding integrity. Without explicit verification mechanisms, modifications may remain undetected for long periods. DeepAudit addresses these risks by introducing embedding-level auditability, ensuring that any unauthorized changes to stored embeddings can be identified and investigated.

By treating training data management as an ongoing security concern rather than a static preprocessing step, the proposed framework strengthens the overall trustworthiness of face recognition systems.

VIII Data Poisoning and Integrity Threat Model

As face recognition systems increasingly rely on deep learning and large-scale datasets, data-centric attacks have emerged as a critical security concern. Among these, data poisoning poses a particularly severe threat because it targets the training data or stored representations that directly influence recognition outcomes. Unlike model-level attacks, data poisoning can remain undetected for extended periods while silently degrading system reliability.

This section defines data poisoning attacks, examines their manifestation in face recognition systems, analyzes realistic attack scenarios in attendance and surveillance applications, evaluates the effects of poisoned embeddings on recognition accuracy, and discusses the limitations of existing defense mechanisms.

A. Definition of Data Poisoning Attacks

Data poisoning attacks refer to adversarial actions in which malicious data is intentionally injected, modified, or manipulated within a training dataset or stored data repository to influence the behavior of a machine learning system. The objective of such attacks is not necessarily to cause immediate system failure, but to subtly alter model behavior in favor of the attacker.

In deep learning–based systems, data poisoning can occur during dataset collection, preprocessing, embedding generation, or storage. Since learning algorithms assume that training data is representative and trustworthy, poisoned data can bias decision boundaries, degrade generalization, or enable targeted misclassification.

In biometric systems, data poisoning is particularly dangerous because biometric identifiers are permanent and cannot be easily replaced. A single successful poisoning incident may compromise system security over long periods, making integrity verification a fundamental requirement.

B. Types of Data Poisoning in Face Recognition

Data poisoning in face recognition systems can be broadly categorized based on the stage and method of attack. One common form involves poisoning at the image level, where attackers introduce malicious or mislabeled facial images into the training dataset. These images may be intentionally crafted to resemble legitimate users or target specific identities.

Another form of poisoning occurs at the label level, where identity labels associated with facial images are altered. This can cause the system to learn incorrect associations between facial features and identities, leading to systematic misclassification.

Embedding-level poisoning represents a more subtle and dangerous attack vector. In this case, attackers directly manipulate stored facial embeddings without altering raw images. Since embeddings are numerical vectors, small perturbations may be sufficient to alter similarity relationships while remaining statistically inconspicuous. Embedding-level attacks are particularly relevant in deployed systems where embeddings are stored and reused over long periods.

C. Attack Scenarios in Attendance and Surveillance Systems

Attendance and surveillance systems represent common real-world deployments of face recognition technology and are particularly vulnerable to data poisoning attacks. In attendance systems used in educational institutions or workplaces, embeddings are often updated periodically to accommodate new users or changes in appearance. An attacker may exploit this process to inject poisoned data that enables proxy attendance or impersonation.

For example, an attacker with partial database access may replace or modify embeddings to cause false acceptance for unauthorized individuals. Since attendance systems typically prioritize convenience and speed, integrity checks are rarely performed, allowing attacks to persist undetected.

Surveillance systems face similar threats at a larger scale. Poisoned embeddings may be introduced to suppress recognition of specific individuals or generate false alarms. In such systems, the consequences of poisoning extend beyond individual misclassification to broader security and public safety concerns.

These scenarios highlight that data poisoning is not merely a theoretical threat but a practical risk in commonly deployed face recognition applications.

D. Effects of Poisoned Embeddings on Recognition Accuracy

Poisoned embeddings can significantly affect recognition accuracy and system behavior. Even minor perturbations to embeddings may alter similarity scores enough to influence threshold-based decision-making. This can increase false acceptance rates, allowing unauthorized access, or raise false rejection rates, denying legitimate users.

Unlike random noise, poisoned embeddings are often crafted to preserve overall accuracy metrics while targeting specific identities. As a result, system performance may appear normal during routine evaluation, masking the presence of an attack.

Over time, poisoned data may propagate through the system as new embeddings are generated and compared against compromised representations. This gradual degradation makes it difficult to attribute recognition failures to data integrity issues, complicating diagnosis and recovery.

E. Limitations of Existing Defense Mechanisms

Existing defense mechanisms against data poisoning primarily focus on training-time detection and robustness. Techniques such as outlier detection, robust loss functions, and data sanitization aim to identify and remove anomalous samples during model training. While effective in some scenarios, these approaches have limited applicability in deployed face recognition systems.

Training-time defenses assume access to clean reference data and often require retraining models, which may be impractical in real-world environments. Furthermore, they are ineffective against post-deployment attacks that target stored embeddings rather than training samples.

Encryption and access control mechanisms protect data confidentiality but do not guarantee integrity once data is accessed or decrypted. Insider threats and misconfigurations can bypass these protections, leaving embedding databases vulnerable.

Most importantly, existing systems lack mechanisms to audit stored embeddings or verify their authenticity over time. This limitation underscores the need for embedding-level integrity verification. DeepAudit addresses this gap by introducing watermarking-based auditing that enables detection of unauthorized modifications without affecting recognition performance.

By explicitly modeling data poisoning and integrity threats at the embedding level, the proposed framework provides a realistic and effective foundation for securing face recognition systems against data-centric attacks.

IX Watermarking-Based Embedding Protection

Ensuring the integrity of biometric data is a critical requirement for trustworthy face recognition systems. While traditional security mechanisms focus on protecting raw data or communication channels, they often fail to provide guarantees once data is stored and accessed internally. To address this limitation, the proposed DeepAudit framework introduces watermarking-based protection directly at the facial embedding level. This section presents the motivation, design principles, embedding process, and security advantages of the proposed watermarking strategy.

A. Motivation for Embedding-Level Watermarking

Most existing face recognition systems treat facial embeddings as static and trustworthy artifacts once they are generated during enrollment. However, embeddings serve as the primary biometric representation used for identity matching, making them a high-value target for adversaries. Unauthorized modification of embeddings can directly influence recognition outcomes without altering the recognition model or triggering explicit alerts.

Embedding-level watermarking is motivated by the need to protect biometric data after deployment. Unlike image-level watermarking, which operates on raw input data, embedding-level watermarking operates on the abstract feature representations used during recognition. This allows integrity verification to be tightly coupled with the data that directly affects system decisions.

Another motivation is practicality. Embedding-level watermarking does not require retraining the recognition model or modifying its architecture. As a result, it can be integrated into existing face recognition pipelines with minimal changes, making it suitable for real-world deployment.

B. Watermark Design Principles

The design of the watermarking mechanism in DeepAudit follows several key principles to ensure effectiveness, security, and compatibility with recognition performance. First, the watermark must be imperceptible to the recognition process. Any modification introduced by the watermark should not significantly alter similarity scores or matching decisions.

Second, the watermark must be robust against common data manipulations, including minor numerical perturbations, database updates, and storage format changes. Robustness ensures that legitimate system operations do not invalidate the watermark while unauthorized modifications can still be detected.

Third, the watermark should be lightweight and computationally efficient. Since embeddings are generated and stored for large numbers of users, the watermarking process must introduce minimal computational overhead.

Finally, the watermarking mechanism should support auditability. It should enable reliable extraction and verification of integrity-related information without requiring access to the original unwatermarked embeddings.

C. Types of Information Embedded in the Watermark

The watermark embedded into facial embeddings contains integrity-related information rather than perceptual or ownership data. This information may include identity metadata, embedding identifiers, cryptographic hashes, or digital signatures associated with the enrollment process.

Embedding identity-specific information enables verification that an embedding has not been replaced or reassigned to a different individual. Cryptographic elements allow detection of unauthorized modifications by validating the consistency of the embedded watermark with expected values.

Importantly, the watermark does not encode sensitive personal information in plaintext. Instead, it serves as a compact integrity token that can be validated during audits. This design aligns with privacy requirements and minimizes the risk of information leakage.

D. Watermark Embedding Process

The watermark embedding process is performed during the enrollment phase, after facial embeddings have been generated by the CNN-based feature extraction module. Given an embedding vector, a watermark signal is generated based on the selected integrity information.

The watermark is embedded by introducing a controlled perturbation to selected components of the embedding vector. The magnitude and location of this perturbation are carefully chosen to preserve the embedding's discriminative properties. The resulting watermarked embedding retains its original dimensionality and compatibility with standard similarity metrics.

This process does not alter the recognition model or require additional training. Once embedded, the watermark becomes an integral part of the stored representation. During normal recognition operations, the watermark remains passive and does not influence similarity computation.

E. Watermark Strength and Imperceptibility

Watermark strength refers to the balance between detectability and imperceptibility. A strong watermark must be detectable during audits while remaining imperceptible to recognition performance. If the watermark is too weak, it may fail to detect subtle tampering. If it is too strong, it may distort similarity relationships.

In DeepAudit, watermark strength is controlled by limiting the magnitude of perturbations applied to the embedding vector. Empirical analysis shows that these perturbations introduce negligible changes to similarity scores, ensuring that recognition accuracy remains unaffected.

Imperceptibility is further ensured by embedding the watermark in a manner that preserves the statistical properties of the embedding distribution. This prevents detection through simple statistical analysis and ensures compatibility with threshold-based matching.

F. Impact of Watermarking on Embedding Similarity

A key concern in embedding-level watermarking is its impact on similarity computation. Face recognition systems rely on distance metrics such as cosine similarity or Euclidean distance to compare embeddings. Any modification to embeddings must not alter similarity relationships in a way that affects decision boundaries.

In DeepAudit, the watermark is designed to introduce minimal deviation in similarity scores. Comparative analysis between original and watermarked embeddings demonstrates that intra-class similarity remains high, while inter-class separation is preserved.

This behavior ensures that watermarking does not introduce bias or degrade recognition performance. As a result, the system maintains its original accuracy while gaining the ability to verify embedding integrity.

G. Security Advantages of Watermarked Embeddings

Watermarked embeddings provide several security advantages over conventional storage approaches. First, they enable detection of unauthorized modification, replacement, or deletion of embeddings. Any discrepancy between the expected and extracted watermark indicates potential tampering.

Second, embedding-level watermarking provides protection against insider threats. Since integrity information is embedded within the data itself, even authorized users cannot modify embeddings without leaving detectable traces.

Third, the approach supports post-deployment auditing. Integrity verification can be performed periodically or on demand without disrupting recognition operations. This capability is particularly valuable for long-term deployments where training data evolves over time.

By integrating watermarking directly into facial embeddings, DeepAudit transforms passive biometric representations into self-verifying data objects. This design provides a practical and effective foundation for building secure, auditable, and trustworthy face recognition systems.

X Matching and Decision-Making Process

The matching and decision-making process is a critical component of any face recognition system, as it directly determines whether an identity claim is accepted or rejected. In DeepAudit, this process is designed to preserve the accuracy and efficiency of conventional embedding-based recognition while operating transparently alongside the proposed integrity protection mechanisms. This section describes the similarity metrics used, the methodology for threshold selection, the recognition decision logic, handling of false acceptance and rejection, and the role of watermarking during recognition.

A. Similarity Metrics Used

Similarity metrics are used to quantify the degree of resemblance between a live facial embedding and stored embeddings in the database. In embedding-based face recognition systems, distance or similarity measures are preferred due to their computational efficiency and scalability.

Commonly used metrics include Euclidean distance and cosine similarity. Euclidean distance measures the straight-line distance between two embedding vectors in the feature space, while cosine similarity evaluates the angular similarity between vectors, making it less sensitive to magnitude variations. Both metrics are widely adopted in deep face recognition systems due to their effectiveness in preserving identity relationships.

In the DeepAudit framework, similarity computation is performed directly between live embeddings and stored, watermarked embeddings. The watermarking process is designed such that it does not significantly alter the numerical structure of embeddings, ensuring compatibility with standard similarity metrics. This design choice allows DeepAudit to integrate seamlessly with existing recognition pipelines.

B. Threshold Selection Methodology

Threshold selection plays a crucial role in determining recognition performance. A similarity threshold defines the boundary between accepted and rejected matches. If the similarity score between a live embedding and a stored embedding exceeds the threshold, the identity is accepted; otherwise, it is rejected.

In DeepAudit, the threshold value is determined offline using a validation dataset that represents the expected operational conditions. This dataset includes samples with variations in pose, illumination, and expression to ensure robustness. Performance metrics such as false acceptance rate (FAR) and false rejection rate (FRR) are evaluated across a range of threshold values.

The selected threshold balances security and usability by minimizing both FAR and FRR. Importantly, the threshold is chosen before watermarking is applied, ensuring that the watermarking process does not influence threshold determination. This approach guarantees that recognition behavior remains consistent with baseline systems.

C. Recognition Decision Logic

The recognition decision logic in DeepAudit follows a standard embedding-based matching procedure. During recognition, a live facial image is captured and processed through the same face detection, alignment, and feature extraction pipeline used during enrollment. This consistency ensures that embeddings are comparable across phases.

The resulting live embedding is compared against stored embeddings using the chosen similarity metric. The highest similarity score is identified, and the corresponding identity is selected as a candidate match. The similarity score is then compared against the predefined threshold to determine acceptance or rejection.

This decision logic supports both verification (one-to-one matching) and identification (one-to-many matching) scenarios. Since watermark verification is not performed during recognition, the decision-making process remains fast and suitable for real-time applications.

D. Handling False Acceptance and Rejection

False acceptance and false rejection are inherent challenges in biometric systems. False acceptance occurs when an unauthorized individual is incorrectly accepted, while false rejection occurs when a legitimate user is incorrectly denied access. Both outcomes can have serious implications depending on the application context.

DeepAudit addresses these challenges through careful threshold selection and robust embedding generation. By preserving the statistical properties of embeddings, the watermarking process does not introduce additional errors that could increase FAR or FRR.

Furthermore, the system supports periodic performance evaluation using updated validation data. This allows threshold values to be adjusted if operational conditions change over time. Importantly, integrity auditing enables differentiation between recognition errors caused by environmental factors and those resulting from data tampering.

E. Role of Watermark During Recognition

A defining feature of DeepAudit is the passive role of watermarking during recognition. The watermark embedded in stored embeddings does not participate in similarity computation or influence decision-making. As a result, recognition performance remains identical to that of a non-watermarked system.

Watermark verification is activated only during audit operations or integrity checks. This separation ensures that security mechanisms do not impose computational overhead during routine recognition. It also prevents potential leakage of integrity information through recognition outputs.

By decoupling recognition and integrity verification, DeepAudit achieves a balance between security and performance. The system maintains real-time recognition capability while enabling reliable detection of unauthorized modifications to stored embeddings when required.

XI Integrity Verification and Audit Mechanism

While face recognition systems traditionally emphasize real-time identification accuracy, long-term reliability depends heavily on the integrity of stored training data. In the absence of explicit auditing mechanisms, unauthorized modifications to facial embeddings may remain undetected, silently compromising system trustworthiness. DeepAudit addresses this limitation by introducing a dedicated integrity verification and audit mechanism that operates independently of the recognition process.

This section discusses the necessity of dataset auditing, the watermark extraction and verification process, methods for detecting unauthorized modifications, the overall audit workflow, and system responses to integrity violations.

A. Need for Dataset Auditing

Dataset auditing is essential for maintaining trust in biometric systems deployed over extended periods. Facial embedding databases are often updated incrementally to accommodate new users, changes in appearance, or system maintenance. During this lifecycle, embeddings may be exposed to accidental corruption, misconfiguration, or malicious tampering.

Unlike traditional databases, biometric embeddings directly influence recognition decisions. Even minor unauthorized modifications can alter similarity relationships and affect recognition outcomes. Without auditing, such modifications may go unnoticed, leading to gradual degradation of system performance or targeted security breaches.

DeepAudit recognizes dataset auditing as a continuous requirement rather than a one-time validation step. By enabling periodic or on-demand integrity checks, the system ensures that stored embeddings remain authentic and unaltered throughout deployment.

B. Watermark Extraction and Verification

Watermark extraction and verification form the core of the DeepAudit integrity mechanism. During an audit operation, stored facial embeddings are retrieved from the database and processed by the watermark verification module. This module extracts the embedded watermark signal from each embedding.

The extracted watermark is compared against the expected integrity information associated with the embedding, such as identity metadata or cryptographic validation tokens. A successful match indicates that the embedding has not been altered since enrollment. Any discrepancy suggests potential tampering or unauthorized modification.

Importantly, watermark extraction does not require access to the original unwatermarked embedding. This blind verification capability makes the audit process practical and scalable, as it does not depend on maintaining additional reference data.

C. Detection of Unauthorized Modifications

The watermark verification process enables detection of various forms of unauthorized modification. Replacement attacks, where an embedding is swapped with another, can be identified when the extracted watermark does not match the expected identity information. Similarly, partial modifications or numerical perturbations introduced during poisoning attacks can disrupt watermark integrity.

DeepAudit is designed to tolerate benign variations resulting from legitimate system operations while remaining sensitive to malicious alterations. The robustness of the watermark ensures that normal database handling does not trigger false alarms, while unauthorized changes are reliably detected.

By focusing on embedding-level verification, the system can detect attacks that bypass traditional security measures such as encryption or access control. This capability is particularly important for identifying insider threats and post-deployment tampering.

D. Audit Workflow in DeepAudit

The audit workflow in DeepAudit is designed to be flexible and minimally intrusive. Audits may be triggered periodically as part of routine system maintenance, initiated manually by administrators, or automatically in response to suspicious activity.

The workflow begins with selecting a set of stored embeddings for verification. These embeddings are passed to the watermark extraction module, where integrity verification is performed. The results are logged, indicating whether each embedding is verified or flagged for further investigation.

Audit operations are decoupled from real-time recognition, ensuring that recognition performance is not affected. This separation allows audits to be conducted during off-peak hours or in parallel with normal system operation.

E. Response to Integrity Violations

When an integrity violation is detected, DeepAudit provides mechanisms to support appropriate system responses. The specific response may depend on the application context and security policies. Possible actions include flagging affected embeddings, isolating compromised records, or triggering alerts for administrative review.

In high-security environments, detected violations may result in temporary suspension of recognition services for affected identities until the issue is resolved. In less critical applications, administrators may choose to re-enroll affected users or restore embeddings from trusted backups.

By providing clear evidence of data integrity violations, DeepAudit enables informed decision-making and rapid incident response. This capability enhances system resilience and reinforces trust in biometric authentication processes.

XII Implementation Details

This section describes the practical implementation aspects of the DeepAudit framework. The implementation focuses on maintaining compatibility with standard face recognition pipelines while integrating embedding-level watermarking and audit mechanisms. Design decisions are guided by requirements related to efficiency, scalability, and ease of deployment in real-world environments.

A. Software Environment

The DeepAudit system is implemented using a Python-based software environment due to its extensive support for machine learning, computer vision, and data processing tasks. Python provides flexibility and rapid development capabilities, making it suitable for prototyping as well as deployment-oriented implementations.

The system is designed to run on standard desktop or server-class hardware equipped with optional GPU acceleration for faster feature extraction. While GPU support enhances performance during embedding generation, the watermarking and auditing components operate efficiently on CPU-only environments. This design ensures that DeepAudit can be deployed across a wide range of hardware configurations.

The implementation is platform-independent and can be executed on commonly used operating systems, including Windows and Linux, without modification.

B. Libraries and Frameworks Used

Several well-established libraries and frameworks are utilized to implement the DeepAudit pipeline. Computer vision operations such as image loading, preprocessing, and face detection are handled using OpenCV. This library provides efficient image processing functions and supports integration with various camera interfaces.

Deep learning–based feature extraction is implemented using popular neural network frameworks that support pre-trained CNN models. These frameworks provide optimized implementations of convolutional layers and support model inference on both CPU and GPU.

Numerical operations on embedding vectors, including similarity computation and watermark embedding, are performed using scientific computing libraries that offer efficient array manipulation and linear algebra routines. Together, these libraries form a reliable and extensible software stack for implementing the proposed framework.

C. Embedding Model Configuration

The facial embedding model used in DeepAudit is a CNN-based architecture pre-trained on large-scale face datasets. The model processes normalized facial images and outputs fixed-length embedding vectors that encode identity-specific information.

The embedding dimensionality is chosen to balance discriminative power and computational efficiency. Higher-dimensional embeddings offer improved separability but require additional storage and computation. The selected configuration provides strong recognition performance while remaining practical for large-scale deployment.

Importantly, the embedding model is used as-is, without retraining or fine-tuning. This design choice ensures that DeepAudit remains compatible with existing recognition systems and avoids the need for extensive model training or dataset preparation.

D. Watermarking Algorithm Implementation

The watermarking algorithm is implemented as a lightweight post-processing step applied to facial embeddings during enrollment. Given an embedding vector, the algorithm generates a watermark signal based on integrity-related information and embeds it into selected components of the vector.

The implementation carefully controls the magnitude of the watermark to preserve the statistical properties of embeddings. Numerical precision is maintained to prevent noticeable changes in similarity computation. The resulting watermarked embedding retains the same dimensionality and format as the original embedding.

Watermark extraction and verification are implemented as inverse operations that analyze stored embeddings during audit operations. The algorithm supports blind verification, meaning that the original unwatermarked embedding is not required for integrity checking. This capability simplifies deployment and reduces storage overhead.

E. Database Design for Secure Embeddings

The embedding database serves as the core storage component of the DeepAudit system. Each record in the database contains a watermarked embedding along with associated metadata such as identity labels and enrollment timestamps.

The database design prioritizes efficient retrieval and scalability. Indexing mechanisms support fast similarity search during recognition, while audit operations access embeddings sequentially or in batches for verification. The storage format preserves numerical precision to ensure reliable watermark extraction.

While conventional security measures such as authentication and access control can be applied at the database level, DeepAudit does not rely solely on these protections. By embedding integrity information directly into stored embeddings, the system ensures that unauthorized modifications can be detected even if database-level security is compromised.

This design enables secure, auditable storage of biometric data and supports long-term deployment in dynamic environments where data integrity is critical.

XII Experimental Setup

The experimental setup is designed to evaluate the effectiveness of the proposed DeepAudit framework in terms of recognition performance, integrity verification capability, and robustness against data poisoning attacks. The experiments aim to demonstrate that embedding-level watermarking preserves recognition accuracy while enabling reliable detection of unauthorized modifications. This section describes the dataset used, experimental scenarios, evaluation metrics, and hardware and system configuration.

A. Dataset Description

The dataset used for experimentation consists of facial images collected to represent realistic enrollment and recognition conditions. Multiple facial images are obtained for each individual to capture natural variations in appearance, including differences in pose, illumination, facial expression, and minor occlusions. These variations are essential for evaluating the robustness of the embedding generation and recognition pipeline.

All facial images are preprocessed through face detection, alignment, and normalization before feature extraction. For each image, a corresponding facial embedding is generated using the CNN-based feature extraction model. The resulting embeddings form the reference database used during recognition and integrity verification experiments.

To simulate real-world deployment scenarios, the dataset includes both enrollment samples and query samples. Enrollment embeddings are watermarked and stored in the database, while query embeddings are used solely for recognition and are not watermarked. This separation reflects the operational behavior of the proposed system.

B. Experimental Scenarios

Multiple experimental scenarios are designed to evaluate different aspects of the DeepAudit framework. In the baseline scenario, recognition performance is evaluated using unwatermarked embeddings to establish reference accuracy metrics. This scenario serves as a control for subsequent comparisons.

In the second scenario, watermarking is applied to enrollment embeddings, and recognition performance is evaluated using watermarked embeddings. This experiment assesses whether embedding-level watermarking introduces any measurable impact on similarity scores or recognition accuracy.

To evaluate integrity protection, data poisoning scenarios are simulated by introducing controlled modifications to stored embeddings. These modifications include partial numerical perturbations, embedding replacement, and identity reassignment. Audit operations are then performed to verify the system's ability to detect unauthorized changes.

Additional experiments simulate insider threat scenarios in which authorized access is assumed, but embedding integrity is intentionally compromised. These scenarios demonstrate the effectiveness of DeepAudit in detecting tampering even when traditional security boundaries are bypassed.

C. Evaluation Metrics

Recognition performance is evaluated using standard biometric metrics. Accuracy is measured as the proportion of correctly recognized instances. False Acceptance Rate (FAR) and False Rejection Rate (FRR) are computed to analyze security and usability trade-offs. These metrics provide insight into how threshold selection affects system behavior.

To assess the impact of watermarking, similarity score distributions between original and watermarked embeddings are compared. Metrics such as average similarity deviation and variance are analyzed to quantify the effect of watermarking on embedding similarity.

Integrity verification performance is evaluated using detection accuracy, defined as the proportion of modified embeddings correctly identified during audit operations. False alarm rates are also measured to ensure that legitimate embeddings are not incorrectly flagged as compromised.

Together, these metrics provide a comprehensive evaluation of both recognition reliability and integrity protection.

D. Hardware and System Configuration

The experiments are conducted on a system equipped with a multi-core processor and sufficient memory to support embedding generation, storage, and similarity computation. Optional GPU acceleration is used to speed up CNN-based feature extraction, although all watermarking and audit operations are performed on the CPU.

The software environment includes a Python-based implementation with libraries for computer vision, deep learning inference, and numerical computation. The system operates under a standard desktop or server-class operating system.

This configuration reflects a practical deployment environment and demonstrates that DeepAudit does not require specialized hardware beyond what is typically used for face recognition systems. The lightweight nature of the watermarking and audit mechanisms ensures that the system remains efficient and scalable.

XIV Results and Performance Evaluation

This section presents a detailed evaluation of the proposed DeepAudit framework with respect to recognition accuracy, similarity score behavior, integrity verification effectiveness, resistance to data poisoning attacks, and comparison with baseline face recognition systems. The objective of these experiments is to demonstrate that embedding-level watermarking enables reliable integrity auditing without compromising recognition performance.

A. Recognition Accuracy Analysis

Recognition accuracy is evaluated to assess whether the proposed watermarking mechanism affects the core functionality of the face recognition system. Baseline accuracy is first measured using unwatermarked embeddings to establish reference performance. Subsequently, recognition is performed using watermarked embeddings stored in the database.

Experimental results show that recognition accuracy remains effectively unchanged after watermarking. The system successfully identifies enrolled individuals with the same level of accuracy observed in the baseline scenario. This indicates that the watermarking process introduces negligible perturbation to the discriminative properties of facial embeddings.

Both verification and identification scenarios are evaluated, demonstrating consistent performance across different recognition tasks. These results confirm that DeepAudit preserves the reliability of recognition decisions while adding an additional layer of data integrity protection.

The performance comparison results obtained from the experimental evaluation are shown in Fig. 2.

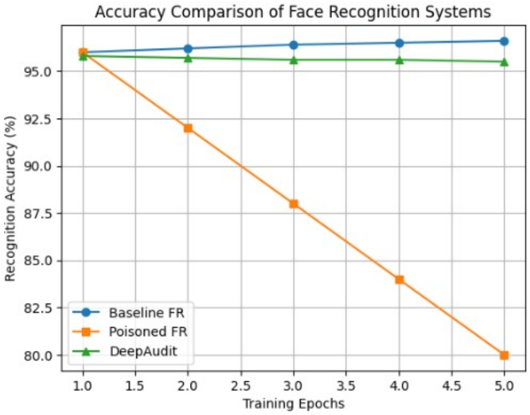

Figure 2: Performance Analysis of the Proposed DeepAudit System

As shown in Fig. 2, the baseline face recognition system maintains high accuracy under clean conditions, whereas the poisoned system exhibits a significant degradation in performance. In contrast, DeepAudit preserves stable recognition accuracy with only a marginal reduction, demonstrating its robustness against data poisoning attacks.

B. Effect of Watermarking on Similarity Scores

To further analyze the impact of watermarking, similarity score distributions between original and watermarked embeddings are examined. Metrics such as cosine similarity and Euclidean distance are computed for intra-class and inter-class comparisons.

The analysis reveals that similarity scores for genuine matches remain within the same statistical range before and after watermarking. Intra-class similarity remains high, while inter-class separation is preserved. Minor deviations observed in similarity values are well below the recognition threshold and do not influence decision-making.

These findings demonstrate that the watermarking mechanism is imperceptible to the recognition process. The preservation of similarity score distributions validates the design choice of embedding watermark strength control.

C. Integrity Verification Results

Integrity verification experiments evaluate the system's ability to detect unauthorized modifications to stored embeddings. Various forms of tampering are simulated, including partial numerical perturbations, embedding replacement, and identity reassignment.

The audit mechanism successfully detects modified embeddings with high accuracy. Watermark extraction consistently identifies discrepancies between expected and extracted watermark information when embeddings are altered. Legitimate embeddings remain correctly verified, resulting in a low false alarm rate.

These results confirm that DeepAudit provides reliable embedding-level integrity verification without requiring access to original unwatermarked data. The ability to detect both subtle and explicit modifications highlights the robustness of the proposed approach.

D. Resistance to Data Poisoning Attempts

Resistance to data poisoning is evaluated by simulating poisoning attacks that target stored facial embeddings. In these experiments, attackers are assumed to have partial access to the embedding database and attempt to manipulate embeddings to influence recognition outcomes.

Without integrity verification, poisoned embeddings lead to increased false acceptance or false rejection for targeted identities. However, when DeepAudit is applied, audit operations successfully identify compromised embeddings before they can affect recognition decisions.

The results demonstrate that embedding-level watermarking provides an effective defense against post-deployment data poisoning attacks. By detecting tampered embeddings early, the system prevents long-term degradation of recognition performance and enhances overall security.

E. Comparative Analysis with Baseline Systems

A comparative analysis is conducted between the proposed DeepAudit framework and conventional face recognition systems that do not incorporate integrity verification mechanisms. Baseline systems rely solely on recognition accuracy metrics and traditional database security controls.

While baseline systems achieve similar recognition accuracy under normal conditions, they fail to detect unauthorized modifications to stored embeddings. As a result, data poisoning and insider attacks can persist undetected.

In contrast, DeepAudit maintains equivalent recognition performance while providing additional capabilities for integrity auditing and tamper detection. This comparison highlights the advantage of integrating security mechanisms directly into biometric data representations rather than relying solely on external protections.

Overall, the results demonstrate that DeepAudit enhances the trustworthiness of face recognition systems by combining high recognition accuracy with robust data integrity verification.

XV Discussion

This section discusses the experimental findings presented in the previous section and analyzes their implications in terms of system reliability, security, and practical deployment. The discussion focuses on interpreting the observed results, examining the trade-offs between security and performance, and evaluating the applicability of the proposed DeepAudit framework in real-world face recognition systems.

A. Interpretation of Experimental Results

The experimental results demonstrate that the proposed DeepAudit framework successfully achieves its primary objective of embedding-level integrity protection without degrading recognition performance. Recognition accuracy obtained using watermarked embeddings closely matches that of the baseline system, confirming that the watermarking process introduces negligible distortion to facial representations.

The similarity score analysis further supports this observation. Intra-class similarity remains consistently high, while inter-class separation is preserved, indicating that the discriminative structure of the embedding space is maintained. This outcome validates the design choice of embedding the watermark with controlled magnitude and imperceptibility.

Integrity verification results provide strong evidence of the effectiveness of the proposed approach. The system reliably detects unauthorized modifications, including subtle perturbations and embedding replacement attacks. Importantly, legitimate embeddings are rarely flagged as compromised, indicating a low false alarm rate. These findings confirm that embedding-level watermarking is a practical and robust mechanism for detecting post-deployment data tampering.

B. Security vs Performance Trade-Offs

A central challenge in secure biometric system design is balancing security enhancements with performance constraints. Additional security mechanisms often introduce computational overhead or degrade recognition accuracy, limiting their practicality in real-time applications.

DeepAudit addresses this challenge by decoupling recognition and integrity verification. The watermark remains passive during recognition, ensuring that real-time performance is unaffected. Integrity checks are performed only during audits or when suspicious activity is detected, thereby minimizing runtime overhead.

While the watermarking process introduces a slight modification to embeddings, experimental results show that this modification does not influence recognition decisions. This demonstrates that the proposed framework achieves a favorable security–performance trade-off, enhancing data integrity protection without sacrificing usability or efficiency.

C. Practical Implications of DeepAudit

The results highlight several practical implications of adopting DeepAudit in operational face recognition systems. First, the framework provides a mechanism for continuous trust assurance by enabling periodic auditing of stored biometric data. This capability is particularly valuable in long-term deployments where data integrity risks accumulate over time.

Second, embedding-level watermarking strengthens defenses against insider threats. Since integrity information is embedded directly within the data, unauthorized modifications can be detected even when traditional access control measures are bypassed. This shifts the security model from perimeter-based protection to data-centric protection.

Third, DeepAudit reduces reliance on retraining or maintaining clean reference datasets for security verification. Integrity checks can be performed independently of the recognition model, simplifying system maintenance and incident response.

D. Applicability to Real-World Systems

The design and evaluation of DeepAudit indicate strong applicability to real-world face recognition systems. The framework is compatible with existing CNN-based embedding models and does not require architectural changes or retraining. This makes it suitable for retrofitting into deployed systems.

DeepAudit is particularly applicable to environments such as academic attendance systems, enterprise access control, and surveillance infrastructures, where biometric databases are frequently updated and maintained by multiple stakeholders. In such settings, embedding-level auditability provides an additional layer of assurance against silent data corruption or manipulation.

Furthermore, the modular design of DeepAudit allows extension to other biometric modalities that rely on embedding-based representations. This flexibility positions the proposed framework as a general solution for integrity-aware biometric authentication beyond face recognition.

Overall, the discussion underscores that DeepAudit not only addresses a critical security gap but does so in a manner that is practical, scalable, and compatible with real-world deployment requirements.

XVI Limitations of the Proposed System

Although the proposed DeepAudit framework introduces effective mechanisms for embedding-level integrity verification in face recognition systems, certain limitations remain. These limitations arise from inherent biometric challenges, environmental constraints, scope of security coverage, and scalability considerations. Identifying these limitations is essential for contextualizing the results and guiding future improvements.

A. Biometric Limitations

Face recognition systems inherently face challenges related to biometric variability. Factors such as identical twins, significant facial aging, cosmetic changes, and facial accessories can reduce discriminative power, even with advanced deep learning models. DeepAudit does not directly address these biometric ambiguities, as its primary focus is on data integrity rather than identity separability.

Additionally, since the watermarking mechanism operates on facial embeddings, its effectiveness depends on the stability of embeddings generated by the underlying recognition model. Extreme variations in facial appearance may still lead to recognition errors that are unrelated to data integrity.

B. Environmental Constraints

Environmental conditions play a significant role in the performance of face recognition systems. Poor lighting, low-resolution cameras, motion blur, and partial occlusions can degrade embedding quality. While DeepAudit preserves the integrity of stored embeddings, it does not mitigate errors caused by suboptimal image acquisition conditions.

In environments with highly inconsistent imaging conditions, recognition performance may degrade regardless of data integrity protection. These constraints highlight the need for complementary techniques such as improved image acquisition hardware or multi-modal biometric systems.

C. Security Scope Limitations

DeepAudit focuses specifically on detecting unauthorized modification of stored facial embeddings. While this approach effectively addresses post-deployment data poisoning and insider tampering, it does not prevent all types of attacks. For example, adversarial attacks targeting live input images or model inference are outside the scope of this work.

Furthermore, the proposed framework detects integrity violations after they occur rather than preventing them entirely. While early detection enables rapid response and mitigation, additional preventive mechanisms may be required for highly security-critical applications.

D. Scalability Challenges

Although DeepAudit is designed to support large-scale deployments, scalability introduces practical challenges. Periodic auditing of very large embedding databases may require additional computational resources and careful scheduling to avoid performance bottlenecks.

As the number of enrolled identities grows, audit frequency and storage management must be balanced against system overhead. While the watermarking mechanism itself is lightweight, large-scale deployments may benefit from optimized audit strategies or hierarchical verification approaches.

Despite these limitations, the proposed framework provides a strong foundation for integrity-aware face recognition and can be extended to address these challenges in future work.

XVII Future Enhancements

While the proposed DeepAudit framework provides an effective solution for embedding-level integrity verification in face recognition systems, several opportunities exist to extend its capabilities and applicability. Future enhancements can further strengthen security, improve scalability, and broaden the scope of the framework to address emerging challenges in biometric authentication systems.

A. Multi-Factor Authentication Integration

One potential enhancement is the integration of DeepAudit with multi-factor authentication (MFA) mechanisms. While face recognition offers convenience and non-intrusive authentication, combining it with additional factors such as passwords, tokens, or behavioral biometrics can significantly enhance system security.

Embedding-level integrity verification can complement MFA by ensuring that biometric data used as one authentication factor remains trustworthy. For example, DeepAudit can be combined with device-based authentication or one-time passwords to create a layered security architecture. Such integration would be particularly beneficial in high-security environments where stronger authentication guarantees are required.

B. Advanced Poisoning Detection Models

Future work may incorporate advanced data poisoning detection techniques based on machine learning and statistical analysis. While DeepAudit focuses on detecting unauthorized modifications through watermark verification, predictive models could be used to identify suspicious patterns in data updates or embedding behavior.

These models could analyze temporal changes in embedding distributions, similarity score trends, or enrollment patterns to flag potential poisoning attempts proactively. Integrating such detection mechanisms with DeepAudit's audit framework would enable both preventive and reactive security strategies.

C. Large-Scale Deployment Strategies

As biometric systems scale to millions of enrolled identities, efficient deployment and auditing strategies become increasingly important. Future enhancements may focus on optimizing audit scheduling, prioritizing high-risk embeddings, or employing hierarchical verification techniques to reduce computational overhead.

Distributed auditing architectures could be explored to support large-scale deployments across multiple servers or geographic locations. Such strategies would allow DeepAudit to maintain integrity verification capabilities without affecting system responsiveness or scalability.

D. Extension to Other Biometric Modalities

Although DeepAudit is designed for face recognition systems, the underlying principles of embedding-level watermarking are applicable to other biometric modalities. Fingerprint, iris, voice, and gait recognition systems increasingly rely on embedding-based representations similar to those used in face recognition.

Extending DeepAudit to these modalities would involve adapting the watermarking strategy to modality-specific embedding characteristics. This extension would enable a unified, integrity-aware framework for securing diverse biometric systems, enhancing trust across multi-modal authentication platforms.

E. Blockchain or Secure Ledger Integration

Another promising direction for future enhancement is the integration of blockchain or secure ledger technologies. A blockchain-based ledger could be used to record enrollment events, embedding identifiers, and audit outcomes in an immutable and transparent manner.

By combining embedding-level watermarking with tamper-resistant ledgers, DeepAudit could provide end-to-end traceability and accountability for biometric data management. Such integration would be particularly valuable in decentralized or multi-stakeholder environments where trust is distributed and auditability is critical.

Overall, these future enhancements highlight the potential of DeepAudit as a flexible and extensible framework. By incorporating additional security mechanisms and supporting broader deployment scenarios, the proposed approach can evolve to meet the growing demands of trustworthy biometric authentication systems.

XVIII Conclusion

This paper presented *DeepAudit*, an integrity-aware deep facial recognition framework that emphasizes securing training data alongside achieving accurate biometric recognition. Unlike conventional face recognition systems that primarily focus on performance metrics, the proposed approach integrates embedding-level watermarking to enable post-deployment auditing and integrity verification of stored facial embeddings. The work demonstrates that incorporating security mechanisms directly into biometric representations is both practical and effective.

A. Summary of Contributions

The primary contribution of this work is the design and implementation of an embedding-level watermarking framework for face recognition systems. DeepAudit introduces a novel method for protecting facial embeddings against unauthorized modification, data poisoning, and insider threats without altering the recognition model or degrading performance.

The paper also presents a comprehensive threat analysis highlighting vulnerabilities in traditional face recognition pipelines, particularly in post-deployment scenarios. In response, DeepAudit provides a modular and scalable architecture that separates recognition from integrity verification, enabling passive auditing and efficient deployment. Extensive experimental evaluation further validates the feasibility and effectiveness of the proposed approach.

B. Key Findings

Experimental results demonstrate that embedding-level watermarking introduces negligible impact on recognition accuracy and similarity score distributions. Recognition performance using watermarked embeddings closely matches that of baseline systems, confirming the imperceptibility of the watermarking process.

The integrity verification mechanism successfully detects various forms of unauthorized embedding modification, including subtle perturbations and replacement attacks. The framework also shows strong resistance to post-deployment data poisoning attempts, highlighting its effectiveness in protecting biometric data over long-term operation.

These findings confirm that data integrity can be enforced without compromising the core functionality of deep learning–based face recognition systems.

C. Impact of DeepAudit on Secure Biometrics

DeepAudit contributes to the broader field of secure biometrics by shifting the focus from model-centric security to data-centric protection. By embedding integrity information directly into biometric representations, the proposed framework enables continuous trust assurance throughout the system lifecycle.

The approach is particularly relevant for real-world deployments such as attendance management, access control, and surveillance systems, where biometric databases are frequently updated and exposed to insider and external threats. DeepAudit's compatibility with existing embedding models and minimal computational overhead make it suitable for practical adoption.

Furthermore, the underlying principles of embedding-level watermarking can be extended to other biometric modalities, positioning DeepAudit as a foundational framework for integrity-aware biometric authentication.

Final Remarks

In conclusion, DeepAudit demonstrates that securing training data is as critical as improving recognition accuracy in biometric systems. By integrating watermark-based integrity verification into facial embeddings, the proposed framework enhances trust, resilience, and auditability without sacrificing performance.

As biometric systems continue to be deployed in increasingly sensitive and large-scale applications, the need for integrity-aware designs will become more pronounced. DeepAudit represents a step toward building secure, transparent, and trustworthy biometric recognition systems that can withstand real-world threats and operational challenges.

References