

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

Dr. Thenmozhi T ¹, Abhinaya V ², Divya B ³, Harikrishnan P ⁴, Indresan V ⁵

¹Professor, HOD, Department of CSE hodcse@kgkite.ac.in

²Department of CSE mvkkumar2006@gmail.com

³Department of CSE divibalakrishnan2005@gmail.com

⁴Department of CSE hari20041203@gmail.com

⁵Department of CSE indreshx2@gmail.com KGiSL Institute of Technology, Coimbatore, 641035, India

ABSTRACT

Face recognition systems have been propelled to new heights by the deep learning. You can now see them everywhere in regulating access and policing crowds, monitoring attendance. The majority of studies narrow down to making such systems more precise and solid. However, frankly, the security and integrity of the training data upon which they work is not discussed sufficiently by people. Deep learning systems are heavily reliant on their data, and, therefore, in case an individual alters or poisons the data, the entire machine can silently crash. Credibility flies down the drain. DeepAudit comes at that point. It is a framework that ensures that face recognition data is not tampered and poisoned. As opposed to tampering with the model or altering its learning process, DeepAudit places tiny, imperceptible watermarks in the facial embeddings during enrollment. These cues do not disrupt performance of recognition in any way. However, when you have to verify the integrity of your data or audit your dataset in the future, those watermarks are available. Here is the mechanism: First during enrollment, a neural net converts facial images into numerical representations. DeepAudit watermarks each of them and stores them safely. Then when recognition is required, the system simply compares live faces to such stored embeddings no muss no messing with the watermark. The watermark will not interfere until it is necessary to check data integrity or audit it. The tests indicate that these embedded watermarks do not affect recognition accuracy or similarity scores. Nevertheless, DeepAudit can detect tampering of the data by an individual. The main takeaway? To be serious about biometric security, it is essential that you are concerned about the integrity of data, and not plain accuracy. DeepAudit provides you with a scalable and practical means of ensuring that face recognition systems are trustworthy on the inside.

Keywords: Adversarial Robustness, Identity Verification Systems, Biometric Template Security, Watermark-Based Protection, Integrity-Aware Recognition, Deep Facial Embeddings, and Data Poisoning Mitigation

How to cite this article: Thenmozhi T, Abhinaya V, Divya B, Harikrishnan P, Indresan V., DeepAudit: An Integrity-Aware Deep Facial Recognition System Using Watermarked Embedding...Int J Drug Deliv Technol. 2026; 16(11s): 726-744; DOI: 10.25258/ijddt.16.11s.75

Source of support: Nil.

Conflict of interest: None

INTRODUCTION

Biometric access control systems have become an inseparable part of the security infrastructure of the modern world and are especially needed in the context of data centers, where data protection and reliable identity authentication is a significant issue [1]. Conventional options of access control like passwords, PIN codes, physical access cards have long been considered to restrict the access of digital systems and secure facilities. Nevertheless, these are also associated with a number of security threats, such as unauthorized sharing, copying, and credentialing. Dependability in authentication is even greater when dealing with a large-scale facility such as a data center where many users and administrators are operating within a highly sensitive environment [2]. Consequently, it has led to the introduction of more organizations in biometric authentication technologies where individuals are authenticated based on distinctive physical traits like their fingerprints, iris pattern, or facial features. The methods used before like Eigenfaces and

Fisherfaces paved the way to the area of facial recognition research, and more recent methods based on deep learning like ArcFace, FaceNet and DeepFace have helped the methodology to become much more accurate in recognition. The latest developments in deep learning systems and applications have also contributed to the creation of more accurate and stable biometric systems [3].

Biometric access control systems are also known to enhance accountability and traceability of secure infrastructures besides improving the reliability of authentication. Since the biometric characteristics are naturally associated with a person, such systems enable organizations to have correct records of the activities on access as well as interaction of the user with the critical systems. This is especially useful in the setting where tracking and auditing access to a system is necessary, like data centers, financial institutions, research laboratories, and governmental facilities. Biometric technologies can minimize the risks of an insider threat, identity theft, and unauthorized access by making sure that sensitive resources can be accessed only by authorized individuals [4].

Although biometric authentication systems have its benefits, there are also new security and privacy issues that

come with its implementation. Biometric templates in databases or those obtained during authentication are sensitive personal information. In case such templates are violated, lost, or damaged without permission, it may cause severe security repercussions [5]. Moreover, when manipulated data is introduced into biometric templates or altered data is added to the recognition process, there is a possibility that an unauthorized user can gain access to secured systems. Thus, the security of biometric data and the validity of biometric representations is a crucial need. Data poisoning and backdoor attacks could also be a threat to machine learning models applied to biometric systems. Other studies have also recently indicated that similar attacks can also be susceptible to a federated learning environment with manipulated training data. Loss functions and evaluation metrics studies are used to enhance model robustness, and watermarking methods are suggested to be embedded into deep neural networks to aid the protection of model integrity and ownership [6].

Face recognition is one of the most popular types of biometric technologies which are present in contemporary digital settings. It is widely used in applications in areas including physical and logical access control, surveillance and monitoring systems, identity verification, border protection and academic attendance tracking. Among the most significant benefits of face recognition, it is worth mentioning that it is a non-obstructive technology, and it does not engage the user to act. Due to this convenience, it is usually preferred to other biometric systems like the fingerprint or iris recognition systems [7].

Deep learning technology has allowed improving face recognition systems greatly. The current deep neural networks can learn powerful face representations that can be used even under varying lighting conditions, facial expressions, angles of view, or age. Such abilities enable face recognition systems to work with great accuracy even under natural conditions and not just in the controlled and laboratory setup. Consequently, face recognition systems that are based on learning are being regularly employed in practice [8].

Nevertheless, the security and integrity issues regarding the training data are applied in any face recognition system are still not properly tackled. The majority of face recognition systems also assume that stored facial embeddings and training datasets have been secure and have not been altered. This assumption might not be true in real-life settings since in most cases, biometric databases are updated, shared and administered by many administrators. As the effectiveness of face recognition systems directly relies on the quality and genuineness of training data, the weakening of the dataset can have a direct effect on the reliability and credibility of the systems. Hence, the issue of security and integrity of training data is a significant challenge that should be implemented in the context of contemporary biometric systems [9].

This project is aimed at the enhancement of the security and reliability of face recognition systems. It talks about the development of face recognition technologies and the use of deep learning-based authentication methods combined with the need to protect data integrity in machine learning.

Since biometric systems could be exposed to attacks like data poisoning and unauthorized data manipulation, further systems are necessary to secure biometric representations. The suggested DeepAudit structure will help in solving such issues by increasing the security of the biometric recognition systems by providing a strength in integrity protection systems [10].

Background of Face Recognition Systems

In order to identify or authenticate people based on their unique facial features taken from photos or video frames, face recognition systems try to do this. Most traditional approaches to creating a face recognition system employed pre-created characteristics and statistical models. Two examples of techniques used to accomplish this are Eigenfaces and Fisherfaces, both of which created facial images in lower-dimensional spaces [11]. Other traditional techniques

used to create a face recognition system were texture-based methods like Local Binary Patterns, which capture data about the textures of a person's face [1].

Even though older face recognition systems performed well in very controlled environments, they generally had trouble with real-world examples because of the factors of variable lighting, face orientation, occlusions, and poor image quality [2]. Additionally, since they employed pre-existing features, they could not generalize successfully across very large and very diverse datasets [3].

Modern face recognition systems have begun to utilize learning-based methods for feature extraction, and capture features of people's faces automatically through the use of facial recognition systems [4]. These systems create numerical vector representations of a person's face, called embeddings, to store high-level semantic information about the identity of the individual [5]. The use of embeddings in face recognition systems allows for simple and efficient similarity processing, as well as high-speed identity comparisons across very large databases, making them usable for applications outside of the lab [6].

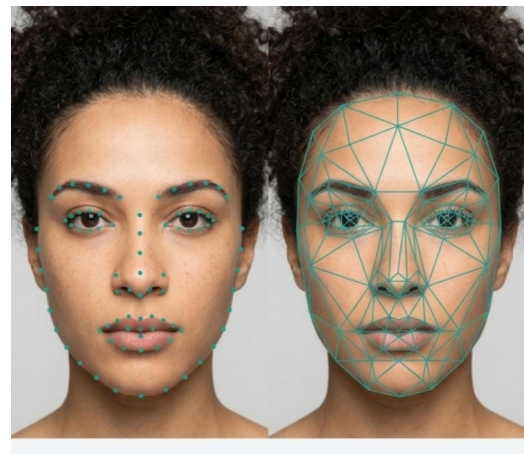


Figure 1: The facial landmarks and geometrical representation of facial structure

Fig. 1 demonstrates how the many variations that each person's face has can be detected and then used as a means

of identification through facial recognition. First, the system will identify facial landmarks representing the significant points of the facial structure (ex. eyes, mouth, nose, jawline) using a series of small yellow dots on the facial image, showing exactly where those significant points would be placed. Once those landmark points have been established, they can then be connected and an additional mesh can be created to represent a geometric structure of the face. This geometric structure constructed from the mesh allows for algorithms to accurately compare each facial pattern to determine identification or verification.

Evolution of Deep Learning in Biometric Authentication

The introduction of deep learning (specifically, convolutional neural networks (CNN)) has greatly impacted the biometric authentication system [7]. CNN's ability to automatically learn hierarchical feature representations directly from raw pixel data (the imagery) means that manual engineering of features is no longer necessary. These capabilities have contributed to improvements regarding the accuracy and robustness of facial recognition systems [8].

Another area where advancements have occurred is with deep embedding models that are created from a facial image by mapping the image to a high dimensional feature space [9]. In this high dimensional feature space, specific embedding representations (i.e. retraining of embeddings) from facial images will be clustered in close proximity to one another while [10] embeddings using different individuals will be spread apart by a significant distance [11]. Deep learning-based biometric systems are becoming increasingly common in important parts of society and the global economy [1], such as those that handle sensitive security data and/or operate on a large scale [2]. But, at the same time, relying on large amounts of training data creates new security vulnerabilities related to the possibility of data manipulation, poisoning, or unauthorized alteration [3].

Importance of Data Integrity in Machine Learning

Data integrity is key to machine learning systems being trustworthy. Training data is what the machine uses to learn the rules (patterns) of the world, so that it can make decisions about the current state of the world [4]. If training data becomes corrupted, compromised, or maliciously modified in some way, then the model will behave in a manner that is no longer predictable and/or way too biased [5].

This kind of issue can cause serious problems in biometric systems, by having the model incorrectly identify people (and create high false acceptances) and/or allowing unauthorized access to people who are not allowed to gain access [6]. Therefore, assuring the authenticity and integrity of both stored training data and derived embeddings is crucial [7]. By utilizing effective integrity verification mechanisms to detect unauthorized modifications and to enable auditability of all data, organizations [8] will significantly increase the level of trust in ML-based biometric systems. [9].

Security Challenges in Biometric Systems

Biometric systems face a wide range of security challenges throughout their lifecycle [10]. Data poisoning attacks involve injecting malicious or misleading samples into training datasets with the intention of degrading recognition performance or causing targeted misclassification [11]. Such attacks can be particularly damaging when executed subtly, as they may evade traditional detection mechanisms [1].

Traditional face recognition systems primarily focus on improving recognition accuracy and efficiency, often overlooking the security of stored embeddings [2]. 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 [3].

Motivation Behind DeepAudit

The industry drive of DeepAudit is the necessity to be no longer accuracy-focused face recognition systems but integrity-conscious biometric systems. Although much has been done to enhance the performance of recognition, relatively little energy has been given to the auditing and security of the training information itself. [4]

DeepAudit fills this gap by proposing an embedding-level watermarking scheme that protects stored facial embeddings against any tampering and unauthorized modification [5]. The system utilizes the concept of post-deployment integrity checking by adding watermarks that are not visible in the face to the facial representation, and does not affect the real-time recognition [6].

Contributions of the Proposed System

The main contributions of the suggested DeepAudit framework consist of the creation of an integrity-conscience face recognition system founded upon the deep facial embedding, the introduction of embedding-level watermarking which defends against tampering and poisoning of training data, the creation of a passive auditing system that does not impact the recognition performance, the analysis of the effect of watermarking on similarity measures and recognition performance, and the focus on making the concept of data security a significant element of designing a biometric system [7].

Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) is a type of deep learning architecture that is based on specialized layers that learn and generate hierarchical features of an image, which are simple edges and then simple forms that become more complex and complex, e.g., facial forms. Within the DeepAudit context, the CNN model, e.g. DeepFace or FaceNet, is the fundamental mapping model that transforms a raw facial image into a numerical embedding [8] This is significant as it allows the system to produce identity-specific representations that are resilient to real-world variations in the form of lighting conditions, pose variations and aging. Once the CNN derives such discriminative facial features, the framework imposes a watermark on the resultant embedding so that the watermark is unaltered, and cannot be data poisoned or compromised [9].

LITERATURE REVIEW

S. No.	Title of the Paper	Objective of the Paper	Paper Outcomes	Limitations
01	A Backdoor Approach With Inverted Labels Using Dirty Label-Flipping Attacks (2024)	To investigate how label-flipping attacks can be used to introduce backdoors into machine learning models trained on external datasets.	Introduced DirtyFlipping, a backdoor attack technique that injects poisoned samples with inverted labels and an audio trigger to manipulate model predictions.	Experiment conducted on limited audio datasets and lacks strong evaluation against advanced defense mechanisms.
02	A Meta-Reinforcement Learning-Based Poisoning Attack Framework Against Federated Learning (2025)	To design an intelligent attack framework capable of generating adaptive poisoning attacks in federated learning environments.	Developed a meta-reinforcement learning poisoning framework that combines CGAN and reinforcement learning to manipulate gradients during model updates.	Computationally expensive and the effectiveness decreases when the number of malicious clients is limited.
03	AIDFL: An Information-Driven Anomaly Detector for Data Poisoning in Decentralized Federated Learning (2025)	To detect malicious client updates in decentralized federated learning systems with non-IID data distributions.	Proposed AIDFL, which uses mutual information, entropy analysis, and clustering techniques to detect anomalous client updates.	Performance degrades when the proportion of malicious clients increases and communication overhead becomes higher.
04	An Optimal Two-Step Approach for Defense Against Poisoning Attacks in Federated Learning (2025)	To develop an effective defense strategy for identifying and filtering poisoned model updates in federated learning.	Introduced TDF-PAD, a two-step detection framework that uses statistical methods such as IQR and Z-score to identify poisoned updates.	Requires a validation dataset and may not perform well in scenarios where labeled validation data is unavailable.
05	Attack on Data: Poisoning and Evasion in E2E Services for Cellular Networks (2025)	To analyze the impact of poisoning and evasion attacks on machine learning models used in cellular network services.	Demonstrated the impact of data manipulation attacks using a real 5G testbed and proposed evaluation metrics for attack assessment.	Study focused on specific network services and does not provide a complete defense mechanism.
06	Attacking-Distance-Aware Attack: Semi-targeted Model Poisoning on Federated Learning (2024)	To design a semi-targeted poisoning attack that manipulates model behavior by exploiting feature-space distance.	Proposed ADA attack that selects optimal target classes based on feature-space distance to increase attack effectiveness.	Existing defenses struggle to detect the attack and the method is mainly evaluated in controlled environments.
07	Data Poisoning Attacks With Hybrid Particle Swarm Optimization Algorithms Against Federated Learning in Connected and Autonomous Vehicles (2023)	To investigate the vulnerability of federated learning models in connected and autonomous vehicles to poisoning attacks.	Developed hybrid particle swarm optimization algorithms (PA-PSOSA and PA-PSOGA) to generate poisoning perturbations in training data.	Experiments conducted mainly on simulated datasets with limited validation in real-world vehicle networks.

08	Coexistence of Deepfake Defenses: Addressing the Poisoning Challenge (2024)	To improve the robustness of deep- fake detection systems against poisoned training datasets.	Proposed a diffusion-based restoration approach using DDPM to remove perturbations from poisoned images and improve detection accuracy.	Method introduces computational overhead and may not generalize well to unseen perturbation patterns.
09	Poisoning Attacks in Federated Learning (2023)	To analyze various types of poisoning attacks in federated learning systems and study their impact on model convergence.	Provided a comprehensive survey of poisoning attack strategies and discussed existing defense mechanisms in federated learning.	Does not introduce a new defense mechanism; focuses mainly on analysis and categorization of existing techniques.
10	Survey on Backdoor Attacks on Deep Learning: Current Trends, Categorization, Applications, Research Challenges, and Future Prospects (2025)	To provide a comprehensive overview of backdoor attacks targeting deep neural networks and identify research challenges.	Categorized backdoor attacks into poisoning and non-poisoning types and analyzed their applications and future research directions.	Survey-based study that does not propose a new defense framework or experimental solution.
11	Embedding Watermarks into Deep Neural Networks (2017)	To protect the intellectual property of trained deep neural network models by embedding digital watermarks.	Demonstrated that watermark information can be embedded into neural network parameters during training without affecting model performance.	Watermarks may become weaker or be removed after extensive model modification such as pruning or fine-tuning.

Table 1: Comparative Analysis of Existing Research on Data Poisoning, Backdoor Attacks, and Watermarking in Machine Learning Systems

Problem Definition and Threat Model

The system of face recognition is dependent on the quality of the training data and the facial embedding in the storage. While modern deep learning models are highly recognizing and feature representing and they generally assume that the training data and stored biometric templates are not altered after being deployed. However, in biometric databases can also be susceptible to unauthorized manipulations, data poisoning, etc. and insider threats which undermine system reliability and trustworthiness. This part technically presents the problem, threat model, and mathematical model of recognition and attack mechanisms.

Problem Definition

The common workflow of conventional face recognition pipelines comprises of face acquisition, feature extraction, and identity matching. These systems give importance to recognition accuracy and efficiency but do not have mechanisms of verifying the accurate storage training data or embeddings [9]. This consequently leads to illegal alterations to stored biometric representations can be invisible and cause wrong authentication. The lack of integrity checking mechanisms enable attackers to modify the stored training samples or embeddings, resulting in a false identification and poor system performance are the consequences [10]. Thus, the integrity that must be ensured and embedded in data.essential prerequisite of safe systems of biometric recognition.



Figure 2: Traditional Face Recognition Workflow

The flow diagram below Fig. 2 demonstrates the rudimentary operation of a conventional face recognition system. The process starts with the capture of an input image and a camera or data. Face detection is then done to identify the position of the system. The facial area on the picture, then features are extracted to find peculiar facial features. Finally, the extracted Embeddings are stored in the database and compared with features to identify the identity of the individual or verify it.

Assumptions in Traditional Face Recognition Systems

The vast majority of face recognition systems have a range of implicit assumptions towards data reliability and system security. Such assumptions can make the system design much easier but could not be true in an adversarial setting where attackers are able use or alter biometric data [11].

Assumption	Description	Risk
Clean Training Data	Dataset contains only valid samples	Vulnerable to poisoning
Secure Storage	Embeddings are not modified	Insider tampering possible
Trusted Environment	No adversarial manipulation	Unrealistic in practice

Table 2: Assumptions in Traditional Face Recognition Systems

The assumptions that are summarized in Table 2 outline significant conditions in which the traditional face recognition is possible systems operate. These systems are typically based on clean training data, safe storage of embeddings and a trusted operational environment. But when such assumptions are not met, then the system can be exposed to attacks, including data poisoning, insider attacks, or adversarial attacks. This leads to system reliability, recognition accuracy, and general security can be highly influenced [1].

Data Poisoning Threat Model

Data poisoning attacks are meant to alter the training data or stored embeddings to manipulate recognition results. Attackers can either inject malicious samples, alter stored embeddings or insert backdoor triggers that trigger targeted misclassification without deterioration of normal system operation. Such attacks take advantage of information weaknesses in model training, model storage, and collection. The goal of the opponent is to decrease recognition accuracy or cause is executed with wrong classification and goes unnoticed when the system is working normally [2].

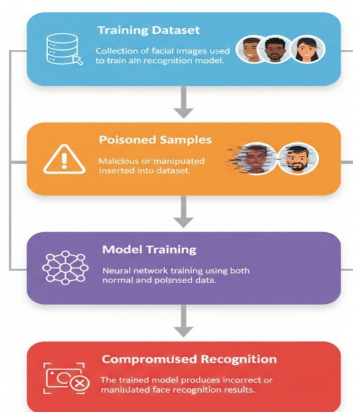


Figure 3: Data Poisoning Attack Architecture

Fig. 3 architecture demonstrates that a face recognition system can be attacked using data poisoning. In this case, bad or corrupted samples are introduced into the training set at the learning stage of the model. These contaminated samples manipulate the acquired feature representations and the model generates erroneous or biased predictions. Subsequently, this can have a major impact on the recognition accuracy and reliability of the system.

Impact of Compromised Training Data

Weakened training information could result in deteriorating performance, high false acceptance and classifier inaccuracy. Polluted samples mislead the distribution of features and decision boundaries resulting in a misaligned identity. approximate and uncontrollable model behaviour [3].

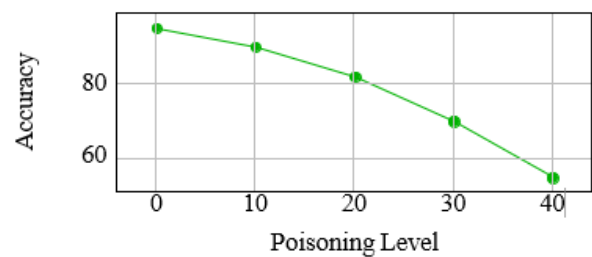


Figure 4: Performance Degradation due to Data Poisoning

The trend of performance depicted in Fig. 4 indicates that the accuracy of a face recognition system is diminishing slowly with the amount of data poisoning. The more the samples poisoned are added to the training set, the more the model develops wrong patterns that result in poor recognition. This degradation underscores the significance of controls that secure the integrity of data, detect against unauthorized things, and guard biometric data against adversarial modifications [4].

Mathematical Formulation of Recognition and Attack Model
 This part constitutes the mathematical description of the recognition algorithm and the model of data poisoning attack, which is the basis of the analysis of the system performance and security.

Face Embedding Representation

Let $n \in \mathbb{R}^n$ represent a facial image. The image is converted to a feature embedding by a convolutional neural network:

$$z = f(n), \quad z \in \mathbb{R}^d \quad (1)$$

The equation (1) suggests that there exists a function $f(n)$ which takes as input an image n and a d -dimensional feature vector (embedding) z , denoting the facial features that are used in recognition [5].

Cosine Similarity

Cosine similarity is used to determine similarity between two embeddings n_i and n_j :

$$S(n_i, n_j)$$

In the equation (2), similarity in the form of the cosine is an evaluation of the similarity between two embeddings

n_i and n_j in form of how close they are comparing the angle of the their vectors.

Cosine similarity is an angular measure of similarity of feature vectors and is popular in embedding-based face recognition systems [3]

Matching Decision Rule

Comparison of identities is done with a threshold τ :

$$Match = \begin{cases} 1, S(n_q, n_d) \geq \tau \\ 0, otherwise \end{cases}$$

The formula (3) shows that n_q is query embedding and n_d is stored embedding. The threshold controls the balance between false rejection and false acceptance [7].

Poisoned Embedding Model

A poisoned embedding can be defined as a perturbation to the original embedding:

$$n' = n + \delta \quad (4)$$

In equation (4) you can find that a poisoned embedding n' is generated by adding a small perturbation δ to the original n where n is the adversarial noise added by the attacker to compromise recognition outcomes.

Embedding Distance

Euclidean distance is used to measure the distance between embeddings:

$$D(n_i, n_j) = \|n_i - n_j\| \quad (5)$$

The Euclidean distance as shown in the equation (5) is a distance measure between two embeddings n_i and n_j feature space.

This mathematical model forms the basis of recognition accuracy analysis and assesses the evaluation of the effects of poisoning attacks in biometrics systems [8].

Proposed DeepAudit Framework

This part introduces the proposed DeepAudit framework that is aimed at providing the integrity and security of facial embeddings applied to recognition systems [9]. The model proposes watermark-based protection of embedding, safe storage, and integrity check, and preserves recognition performance [10].

Overall System Architecture

DeepAudit as a whole consists of the face detection, feature extraction, watermark embedding, secure storage and identity verification modules [11]. The system has two stages; enrollment and recognition. This type of biometric processing using pipelines has found extensive application in the current face recognition systems [1]. The architecture allows biometric data to be processed and handled in a modular fashion and in a secure manner all through the recognition pipeline [2].

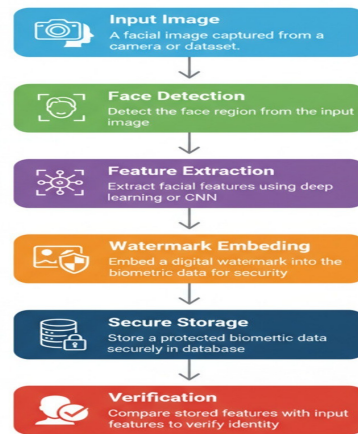


Figure 5: Overall Architecture of DeepAudit Framework

The architecture in Fig. 5 describes the entire DeepAudit pipeline of the face recognition that is secure. The system starts with a facial image acquisition of facial features and extraction to create facial embeddings. These embeddings then a digital watermark is added to prevent integrity and authenticity of the content and then the documents are safely stored. The watermark is also removed and verified during the verification to ensure that there is no interference and unauthorized access to the watermark biometric data.

Modular Design of DeepAudit

DeepAudit is in a modular design where every component is used to perform a separate function. This design improves scalability and system maintainability coupled with allowing secure biometric data processing [3]. To enhance robustness and flexibility in large-scale recognition systems, modular biometric architectures are usually used [4]. Each module is an independent one which allows it to be deployed and the system upgraded flexibly

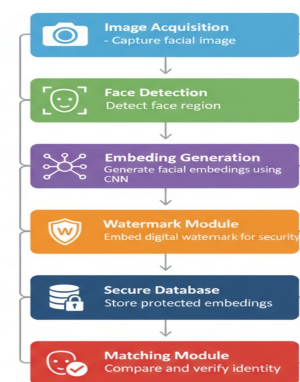


Figure 6: Modular Block Diagram of DeepAudit

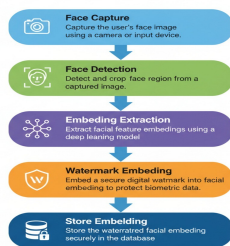
The architecture in Fig. 6 depicts the major elements of the DeepAudit framework using a modular architecture. The system is categorized into modules which include feature extraction, watermark embedding, secure storage and integrity verification. Each of the modules has a particular functionality that provides secure processing and protection

of biometric embeddings. This modular architecture increases the flexibility, scalability and maintainability of the system.

Enrollment Phase

On the enrollment stage, a facial image is captured and fed into a model of deep learning-based feature extraction to produce embeddings [5]. Each embedding should include a watermark which is stored with it to provide integrity and authenticity of stored biometric data [6]. This mechanism makes sure that embeddings stored have integrity information to be verified and guard against tampering in the future [7].

Figure 7: Enrollment Phase Workflow



Face recognition system enrollment stage entails the capturing and registration of the facial information of a user to establish a reference template that will be used in authentication later. This workflow is shown by Fig. 7 and has the steps in a sequence, that is, first the system captures the face image with the help of a camera, and then it processes the image to identify and isolate the facial features. The features are then encoded as digital template and stored safely in the database. The structured workflow will make sure that the system is able to compare live inputs with stored templates later on to enable accurate recognition.

Recognition and Verification Phase

During the recognition stage, a live image is handled and contrasted with embedded embeddings through similarity measures [8]. Before identifying with the identity, integrity verification is done to avoid the use of invalid biometric templates. [9]. Verification also makes sure that only genuine embeddings are involved in recognition to enhance system reliability [10].

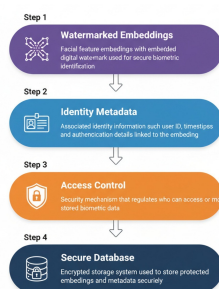


Figure 8: Recognition and Verification Phase Workflow

Recognition and verification stage in a face recognition system entails the identification or verification of people by

reference to the already registered facial templates. This pipeline is depicted in Fig. 8 and it starts with capturing the live. Face image from the user. The image is then fed into the system to identify the face after which appropriate facial features are extracted and compared with the existing templates in the database. According to this comparison, the system either acknowledges the association or authenticity of individual, and secure and verify access control is possible.

Secure Embedding Storage Model

The model of secure storage stores watermarked embeddings and identity metadata [11]. One can restrict access to biometric data to avoid its unauthorized modification and insider attacks [1]. This model increases security against tampering and unauthorized access to biometric templates [2].

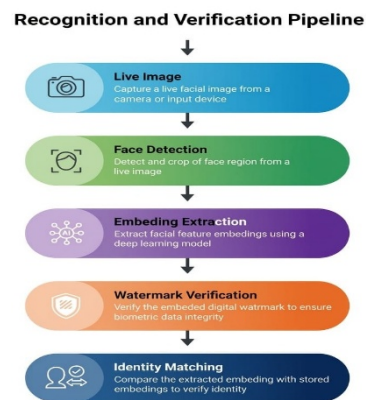


Figure 9: Secure Embedding Storage Architecture

Fig. 9 demonstrates the secure embedding storage architecture that is aimed to defend facial embeddings against access and tampering by unauthorized users. In this design, the facial embedding is encrypted and then it is saved in the database, so that sensitive biometric information is not disclosed. Access control measures are used to determine who can get access and decrypt the embeddings, and integrity measures are used to ensure that the embeddings are not altered or poisoning of the stored data. The design offers an effective structure in order to support privacy and security in face recognition systems.

Algorithm for DeepAudit Workflow

This section presents algorithms for face recognition, embedding generation, and watermark embedding. Algorithmic processing of biometric data and feature embeddings has been widely studied in deep learning-based recognition systems.

Face Recognition Algorithm

The recognition process follows standard embedding-based matching strategies used in modern face recognition systems.

Algorithm 1 Face Recognition Process

- 1: Capture input image I
- 2: Detect face region
- 3: Extract embedding $z = f(I)$
- 4: Verify watermark integrity
- 5: Compute similarity with stored embeddings
- 6: **if** similarity \geq threshold **then**
- 7: Accept identity
- 8: **else**
- 9: Reject identity
- 10: **end if**

Embedding Generation Algorithm

Here is a point where the main contribution of the DeepAudit framework is made, as it suggests protecting facial embeddings with the help of watermarks. The suggested strategy incorporates information about integrity into facial features representations without the impact on recognition. The watermark allows checking the stored embeddings and identifying illegal alterations [3].

Algorithm 2 Embedding Generation

- 1: Input facial image x
- 2: Preprocess image
- 3: Pass through CNN model
- 4: Extract feature vector z 5: Normalize embedding 6: Output embedding z

Watermark Embedding Algorithm

The watermark should fulfill three primary characteristics, namely imperceptibility, resilience and verification of integrity. The embedded watermark is not supposed to have any great impact on similarity scores, should be consistent under normal conditions processing activities, and must be able to detect tampering [4]. The properties are commonly believed to be crucial in the process of secure machine learning and digital watermarking systems [5].

Algorithm 3 Watermark Embedding

- 1: Input embedding E and watermark W
- 2: Select embedding components
- 3: Compute watermarked embedding $E_w = E + \alpha W$
- 4: Store E_w in database

Watermark-Based Embedding Protection

This section presents the core contribution of the DeepAudit framework, which introduces watermark-based protection for facial embeddings. The proposed approach embeds integrity information directly into facial feature representations without affecting recognition performance. The watermark enables verification of stored embeddings and detection of unauthorized modifications [6].

Principles of Watermarking

The watermark is designed to satisfy three main properties: imperceptibility, robustness, and integrity verification capability. The embedded watermark should not significantly alter similarity scores, must remain stable under normal processing operations, and should enable

detection of tampering [7]. These properties are widely considered essential in secure machine learning and digital watermarking systems [8].

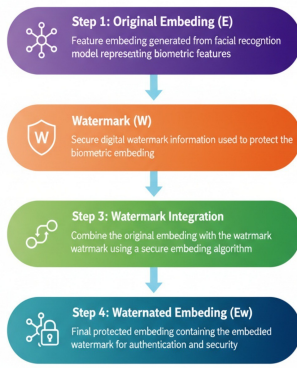


Figure 10: Watermark Model for Embedding Protection

Fig. 10 illustrates the watermark design model. The watermark contains identity metadata or cryptographic signatures and is embedded with controlled strength to preserve recognition accuracy and model performance [9]. This approach ensures that the embedded information can be verified or traced without affecting the underlying facial recognition model, providing both security and accountability in biometric systems.

Embedding-Level Watermark Formulation

Let $E \in R^d$ denote the original facial embedding and $W \in R^d$ denote the watermark vector. The watermark embedding process modifies the original embedding by adding a scaled watermark [10].

$$E_w = E + \alpha W \quad (6)$$

The equation (6) indicates that the watermarked embedding E_w is created by adding a scaled watermark αW to the original embedding E .

where α is a scaling parameter controlling watermark strength. The value of α is chosen to balance robustness and recognition performance [11]. The embedding-level formulation ensures that the watermark is directly integrated into the feature representation without affecting discriminative features learned by deep neural networks [1].

Watermark Embedding Process

Watermark embedding is a procedure that is done at the enrollment stage and then embeddings are stored in the database. This is done to make sure that embedded integrity information is stored in all of the stored embeddings, which allows detection. of interference and illegal alteration.

Digital watermarking is another concept that has been discussed as a way of securing deep learning models and ownership. verification. Uchida et al. suggested an architecture to incorporate watermarks to deep neural networks to safeguard the intellectual right of trained models. Their technique incorporates watermark data in the parameters of neural networks in the training process through parameter regularization approach. The experimental evidence revealed that the embedded watermark can be detected when the model is changed, e.g.

by fine-tuning and even parameter pruning, and yet retaining the original model performance [2].

Fig. 11 shows the steps to watermark embedding that is applied to increase the security and integrity of facial embeddings, and the way in which the watermark is incorporated into facial embeddings without impairing recognition. Accuracy

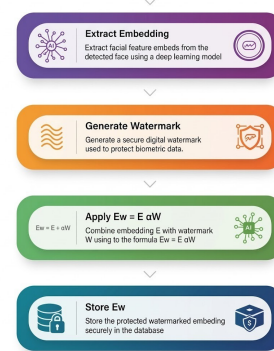


Figure 11: Watermark Embedding Workflow

Watermark Extraction and Verification

To verify integrity, the watermark is extracted from the stored embedding and compared with the original watermark. This verification process supports secure biometric template protection and dataset auditing [3].

The extraction process is defined as:

$$W' = E_w - E \quad (7)$$

The equation (7) indicates that the extracted watermark W' is obtained by subtracting the original embedding E from the watermarked embedding E_w .

Integrity verification is performed using the following condition:

$$V = Valid, (|W - W'| < \epsilon)$$

The equation (8) indicates that the watermark is valid (authentic) if the difference between the original watermark W and extracted watermark W' is less than the threshold ϵ where ϵ is a predefined tolerance threshold. This verification mechanism enables detection of unauthorized embedding modifications and data poisoning attempts [4].

Watermark Strength vs Recognition Accuracy

The watermark robustness and watermark recognition are dependent on parameter α which represents the strength of the watermark. Increasing the value of α enhances the integrity verification capability of the system, but it may reduce similarity scores and recognition performance [5]. A suitable watermark strength ensures that the embedding integrity is preserved without significantly affecting the system's resistance to adversarial manipulation [6].

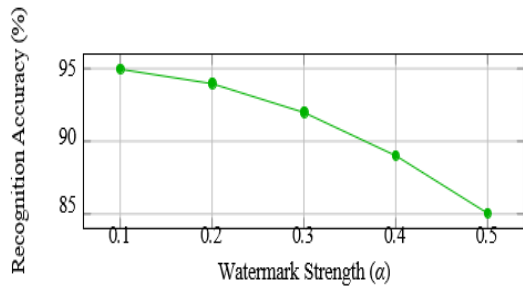


Figure 12: Watermark Strength vs Recognition Accuracy

Fig. 12 shows the trade-off between watermark strength and recognition accuracy. As the watermark strength α increases, the embedded watermark becomes stronger and easier to detect. However, increasing the watermark strength slightly alters the embedding vectors, which may reduce the recognition accuracy of the facial recognition system. Conversely, when the watermark strength is reduced, recognition accuracy improves because the embeddings remain closer to the original values, but the watermark becomes easier to remove or modify. This creates a trade-off between system security and recognition accuracy. Therefore, selecting an optimal watermark strength is essential to maintain both embedding integrity and biometric recognition performance.

Matching and Decision Model

Metric	Formula	Property
Cosine Similarity	$\frac{n_i \cdot n_j}{\ n_i\ \ n_j\ }$	Measures angular similarity
Euclidean Distance	$\ n_i - n_j\ $	Measures absolute distance

Table 3: Comparison of Similarity Metrics

Threshold Selection Strategy

A similarity threshold has been used to make recognition decisions between similar and dissimilar pairs. Validation data is used to find the threshold value in order to strike the right balance between false rejection and false acceptance rates [2]. Proper selection of the threshold is very important in the attainment of optimum recognition and the reliability of the system. A low threshold will raise the probability of acceptance, but can lead to more false acceptance, which lowers the security of the system. On the other hand, high threshold minimises false acceptance but can maximise false rejection, which impacts usability [3]. Consequently, one tuning parameter is threshold tuning where validation datasets are used to determine the optimum operating point.

The matching and decision model identifies identity recognition based on similarity measures and threshold-based classification with the help of facial embeddings. Face images in the face recognition systems based on deep learning, facial images, are modeled as compact feature vectors that represent characteristics that are identity-specific [7]. The procedure is a step of similarity computation, selection of a threshold and the recognition performance measurement with error measures False Acceptance Rate(FAR) and False Rejection Rate(FRR) to name a few.

The incorporation of embedding in recognition has emerged as a common practice in the contemporary biometric systems because it is capable of efficient identity matching in high dimensional feature space [8]. The validity and trustworthiness of however, are doubtful. Proper similarity measures and decision levels are important in recognition decisions. In addition, the error rates should be evaluated in order to make sure that the systems are secure as well as usable in the practice [9]. A. Similarity Metrics

Similarity Metrics

Similarity or distance measures are used to compare the facial embeddings in order to identify correspondence in identity. These features compare a pair of feature vectors of facial features. The most cosine similarity and Euclidean distance are widely applied measures of face recognition [10].

Cosine similarity is an angular distance between embeddings, and it is defined as:

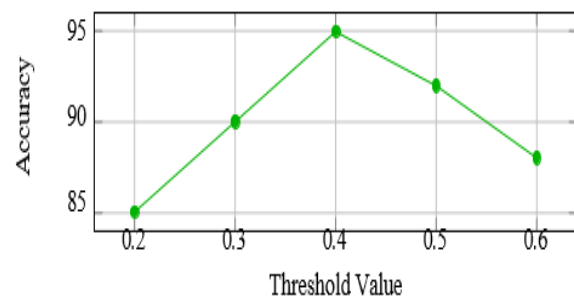


Figure 13: Threshold Tuning for Recognition Decision

Fig. 13 demonstrates how selection of threshold affects the performance of the system. The threshold gives the boundary level where a query input can be identified to match against a stored template. There is an optimum threshold that maximizes accuracy of recognition with minimum false acceptance and rejection, hence the overall system robustness. This is illustrated in the figure that indicates that the accuracy increases to its peak and

decreases as the threshold changes the sensitivity of face recognition systems to threshold adjustment.

Recognition Decision Rule

The identity recognition is done based on comparison of the similarity score and a pre-existing threshold τ . The decision rule will be defined as

$$Match = \begin{cases} 1, & S(n_q, n_d) \geq \tau \\ 0, & otherwise \end{cases}$$

This equation (11) is of a threshold-based decision rule, such that the output is 1 when the similarity $SS(n_q, n_d)$ is more than or equal to threshold.

n_q and n_d are query embedding and stored embedding respectively. The matching is declared when the similarity score is more than the threshold [4]. This rule allows efficient identity checking and matching on large scale large biometric databases.

False Acceptance and False Rejection Analysis

False Acceptance Rate (FAR) and False Rejection Rate (FRR) are used to evaluate the recognition performance and assess the system security and usability [5]. These metrics find extensive application in the biometric authentication to measure the reliability of the system in varying operating environments.

False Acceptance Rate is the likelihood of a false acceptance of an invalid identity

$$FAR = \frac{\text{False Acceptances}}{\text{Total Attempts}}$$

The equation (12) is the False Acceptance Rate (FAR), or a ratio of incorrect acceptance of attempts by the system by unauthorized users.

False Rejection Rate The likelihood of rejecting a genuine identity is given by

$$FRR = \frac{\text{False Rejections}}{\text{Total Attempts}}$$

The equation (13) is an expression of False Rejection Rate (FRR), the ratio of the actual users rejected by the system. A reduced FAR means that there is greater system security

Parameter	Value
Programming Language	Python
Face Detection	OpenCV
Embedding Model	DeepFace
Similarity Metric	Cosine Similarity
Threshold	0.4
Watermark Strength (α)	0.2

Table 4: Implementation Parameters

The main parameters of the implementation of the proposed face recognition and watermarking system are sum-

marized in Table 4. The proposed system is coded in Python, with the usage of OpenCV to detect faces, and DeepFace to

whereas a reduced FRR means that there is better user convenience. The FAR-FRR trade-off is what dictates the overall system performance and reliability of operation of the system at hand. In security-sensitive systems, e.g. access control and identity checking, minimizing FAR is frequently a priority, where user-friendly systems are concerned with minimizing FRR.

Implementation and Experimental Setup

These metrics of evaluation present some (11) ive measurements of analysing recognition accuracy be used to determine the effect of data integrity mechanisms, such as watermark-based embedding protection, on system performance. Experimental Set up and Implementation.

Implementation Environment

In this section, I will detail the environment, characteristics of the data, metrics of evaluation, and layout of the system to be utilized in the evaluation of the proposed DeepAudit framework. The experimental design is aimed to evaluate acceptance performance and entrenchment of integrity in real world settings. The assessment model adheres to the usual biometric recognition systems and security assessment procedures in machine learning-based authentication models [6].

A DeepAudit system is written in Python and is also based on computer vision and deep learning face detection and feature extraction libraries. Python has a more lenient machine learning model deployment and speedy development of systems [7]. Facial representations are generated by a pre-trained CNN-based embedding model that makes it possible to extract discriminative features out of facial images [8]. The module of watermarking is added as a supplementary processing step to insert information on the integrity in feature vectors without altering the recognition model. The system uses OpenCV when it comes to face detection as well as preprocessing such as the alignment and normalisation of pictures. The use of FaceNet as the embedding model is based on the fact that it generates small and discriminative facial representations. closures in higher dimension feature space [9]. Cosine similarity is used in matching identities and a preset level is used to decide on recognition.

extract the embedding. The similarity measure used is the cosine similarity with similarity decision threshold of 0.4 and a watermark strength (α) of 0.2 is selected to trade-off security and accuracy of recognition. The parameters provide efficient feature extraction, immune embedding processing and support with real-time biometric recognition systems [10].

Evaluation Metrics

Recognition accuracy and error-related measurements like False Acceptance Rate (FAR) and False Rejection Rate (FRR) are the measures of the DeepAudit framework performance. These are standard measurement of evaluation with biometric authentication systems to measure the system security and usability [2]. Accuracy of recognition can be defined as

$$Accuracy = \frac{\text{Correct Predictions}}{\text{Total Predictions}} \tag{14}$$

The equation (14) is the accuracy that is the ratio of right predictions of the model against the sum of predictions. False Acceptance Rate is used to calculate unauthorized access acceptance

$$FAR = \frac{\text{False Acceptances}}{\text{Total Attempts}} \tag{15}$$

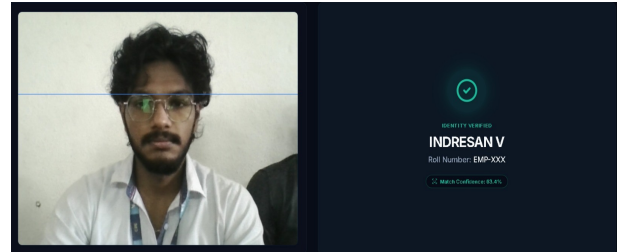
The equation (15) represents the False Acceptance Rate (FAR), which measures the proportion of unauthorized attempts incorrectly accepted by the system.

False Rejection Rate measures incorrect rejection of genuine users

$$FRR = \frac{\text{False Rejections}}{\text{Total Attempts}} \tag{16}$$

Equation (16) is the False Rejection Rate (FRR) which is the rate of the system rejecting the actual users.

Recognition accuracy is used to assess system performance in terms of identification, whereas FAR and FRR assess system reliability in adversarial conditions and under operational constraints [2]. These measures offer quantitative analysis of recognition. precision, strength of



security, and the effects of watermark-based integrity systems [3].

System Configuration

The experiments are performed on an average computing system having moderate hardware facilities that can support real-time processing. The setup proves that the proposed framework can be implemented in an efficient way that does not require. that need specialized hardware acceleration and thus can be used in real biometric applications [4].

Component	Specification
Processor	Intel Core i5
RAM	8 GB
Storage	512 GB SSD
Operating System	Windows 10
GPU	Not Required

Table 5: System Configuration

The experimental system shows that DeepAudit is capable of running in resource-limited systems without compromising on the performance of recognition and implementing integrity checks [5]. Table 5 shows the hardware set up that was used in the experiments and it outlines the elements of the system that were to be used in the real-time processes and secure biometric activities.

Results and Performance Evaluation

This part of the paper analyses the quality of the proposed DeepAudit framework in three aspects (recognition accuracy, watermark impact, integrity verification capability, and data poisoning attack resistance). The experimental results show that the suggested system has a high recognition performance with embedding integrity and security. The assessment is made in accordance with the common practices of assessment of the biometric system and security evaluation employed in authentication machines that utilize machine learning [6].

Recognition Accuracy Analysis

Recognition accuracy is an assessment of the accuracy of the system to identify individuals correctly by use of facial embedding. The large recognition accuracy implies that the

embedding representation maintains identity-related information. even with watermark embedding [7]. Based on experimental assessment, the proposed framework is very accurate, and its embedding security remains intact

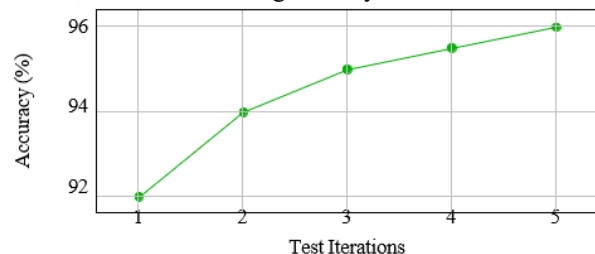


Figure 14: Recognition Accuracy Analysis

The results in The Fig. 14 show that the accuracy is over 95 percent at normal working conditions, which proves that watermark embedding does not have a significant impact on the learning of discriminative features or identity matching capability [8].

Figure 15: Recognition output

The Fig. 15 above shows how DeepAudit can be used to verify a user identity. This output indicates that the system

is capable of successfully authenticating users in real-time maintaining the integrity of the embedded biometric information.

Criteria	Without Watermark	With (Deep-Audit) Watermark
Recognition Accuracy	Slightly higher baseline	Nearly equal (minimal drop)
Data Integrity	No protection	Strong integrity verification
Tamper Detection	Not possible	Detects embedding modification
Resistance to Data Poisoning	Low	High
Security Level	Basic	Enhanced security
Dataset Auditing	Not supported	Supported
Embedding Authenticity	Not guaranteed	Verified using watermark
System Reliability	Moderate	High
Insider Attack Protection	No	Yes
Deployment Trust	Limited	Improved trustworthiness

Table 6: Comparison of Face Recognition Performance With and Without Watermark

Table 6 gives a comparison of the performance of face recognition system with and without DeepAudit watermarking mechanism. Although there is no significant difference in recognition accuracy, DeepAudit watermark increases the integrity of data, allows identifying tampering, rises the resistance to data poisoning, and secures an authenticity of embedding. Greater system reliability, immunity to insider attacks, support of dataset auditing, and general enhancement of deployment trust are also pointed out in the table as benefits of the added security of the watermark with neither negative effects on recognition performance.

Effect of Watermarking on Similarity Scores

The effect of embedding a watermark on feature representations is that the resemblances could change in the determination of similarity scores to provide recognition decisions. Nevertheless, the watermark strength parameter is selected with caution to reduce the impact but at the same time maintaining embedding integrity [9]. The similarity scores are compared in pre watermark embedding and after the embedding is done.

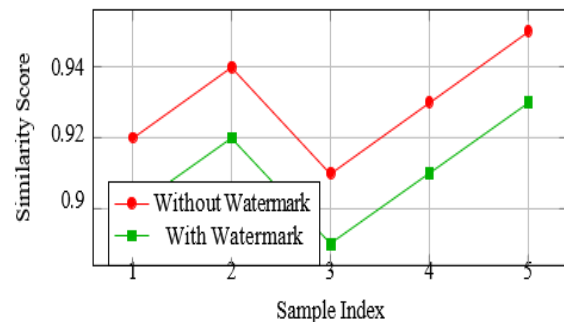


Figure 16: Similarity Score Comparison (With vs Without Watermark)

Figure 16: Similarity Score Comparison (With vs Without Watermark)

Fig. 16 provides a comparison of similarity scores of facial embedding with and without DeepAudit watermarking mechanism. The outcomes indicate that similarity scores differ insignificantly, which proves watermarking is not very significant. recognition of the affect matching decisions. This proves that embedding-level watermarking maintains quality of feature representation and allows integrity checking [10].

Integrity Verification Results

The integrity verification performance is tested based on a confusion matrix of the valid and tampered embeddings. The

confusion matrix is used to quantify success in the system in classifying authentic and modified embeddings correctly.

Actual / Predicted	Valid	Tampered
Valid Embeddings	95	5
Tampered Embeddings	3	97

Table 7: Integrity Verification Confusion Matrix

Table 7 shows the confusion matrix of integrity check of valid and tampered embeddings. The findings show that verification is high with 95/100 verified embeddings being identified and 97/100 verified tampered embeddings

detected. These indicators prove that the integrity verification by the watermark scheme of DeepAudit is capable of effectively identifying the genuine embeddings and the corrupted ones, which provides strong security against integrity attacks.

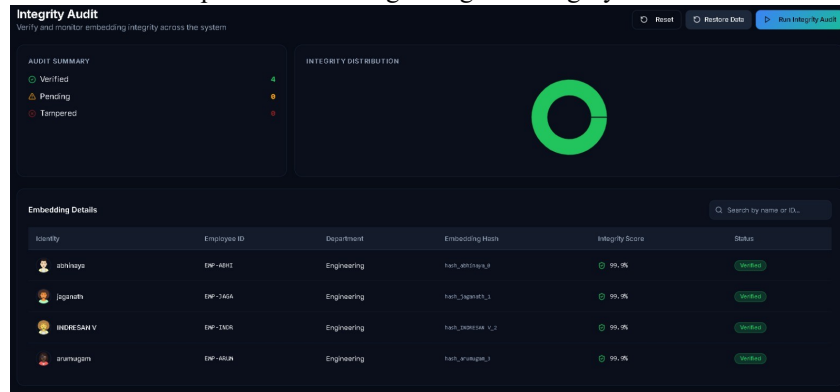


Figure 17: Integrity Verification

Fig. 17 depicts the Integrity Audit dashboard that verifies the success of verification of all the biometric samples. The findings show that there is high accuracy in detecting tampered embeddings which implies effective integrity verification. The watermark removal and verification system proposed effectively identifies any illegal alteration of the watermark and prevents the theft of biometric information [11].

Resistance to Data Poisoning Attacks

To test the system strength against attacks based on data poisoning, the strength of the system is measured by how successful the attacks are when the attacker is undertaken with less than the target data size. raising the levels of poisoning. The goal in data poisoning attacks is to tamper with training data or stored embeddings to affect the outcome of recognition [1].

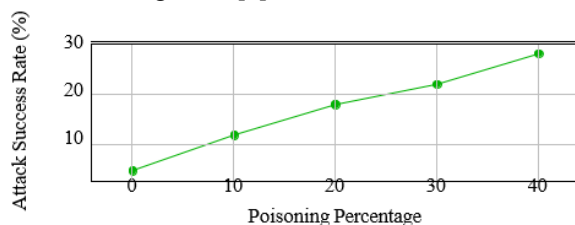


Figure 18: Attack Success Rate under Data Poisoning

Fig. 18 shows the success rate of the attack of the system with various percentages of data poisoning. The higher the percentage of poisoned data, the higher the success rate of attack, which implies that the chances of misclassification or model are greater compromise. Nevertheless, the DeepAudit watermarking and integrity checking systems

reduce these impact and keep the systems robust and unauthorized manipulation of embeddings despite moderate conditions poisoning conditions.

In order to further examine how the manipulation of datasets changes the performance of biometric systems, further experiments that incorporate swapping and label flipping as other forms of adversarial interference in the facial embedding space need to be performed. These manipulations may be a result of untimely changes done by malicious database manipulation, by insider attack, or by antagonistic data injection in a biometric authentication system [2].

These tampering attacks were assessed by comparing the pre and post-assessments of the face recognition system. The findings indicate that the clean dataset has a very high recognition accuracy at low error rates. However, when the dataset is corrupted, the recognition accuracy becomes very low [3]. Moderate degradation of system accuracy is caused by subtle perturbation, whereas embedding swapping and label flipping attacks result into severe performance degradation as a result of false identity affiliations in the embedding space.

Note: The table shows the decrease in the metric of biometric accuracy and error rates when the dataset is exposed to attacks on the feature manipulation, as well as detection ability of the proposed system.

Despite the severe degradation in recognition performance caused by these attacks, the DeepAudit framework

effectively identifies manipulated embeddings using tampering detection rates, demonstrating the effectiveness of watermark-based verification. The system achieves high of embedding integrity protection mechanisms in

	Original Dataset (Before Tampering)	Subtle Perturbation	Embedding Swapping	Label Flipping
Accuracy (%)	99.20 ± 0.15	84.50 ± 1.20	45.30 ± 2.10	32.15 ± 1.80
False Acceptance Rate (FAR)	0.002 (0.2%)	0.154 (15.4%)	0.482 (48.2%)	0.612 (61.2%)
False Rejection Rate (FRR)	0.005 (0.5%)	0.128 (12.8%)	0.515 (51.5%)	0.654 (65.4%)
Equal Error Rate (EER)	0.003 (0.3%)	0.141 (14.1%)	0.498 (49.8%)	0.633 (63.3%)
Precision	0.991	0.832	0.441	0.315
Recall	0.995	0.872	0.485	0.346
F1-Score	0.993	0.851	0.462	0.329
Tampering Detection Rate	N/A	94.5%	99.8%	100.0%

Table 8: Face Recognition Performance Metrics: Before and After Data Tampering

As the Table 8 demonstrates, watermark-based integrity checking decreases the effects of poisoning attacks as such detection blocks the recognition of manipulated embeddings. This shows an enhanced system robustness and resistance to manipulation of adversarial data [5].

Comparative Analysis with Baseline Methods

The suggested DeepAudit framework is contrasted with the basic face recognition methods that lack the implementation of embedding defense strategies. The comparison measures the aptitude of recognition and the availability of integrity protection.

Method	Accuracy (%)	Integrity Protection
Traditional Face Recognition	94	No
CNN-based Recognition	96	No
Proposed Deep Audit	95	Yes

Table 9: Comparison with Baseline Methods

In Table 9, the suggested DeepAudit system is compared to the baseline face recognition systems in respect of accuracy of recognition and integrity assurance. Although traditional and CNN-based methods of recognition are slightly higher precision, they do not have an embedding-level integrity check. DeepAudit offers a similar accuracy with a high level of integrity protection with the use of watermarking, which is its strength in maintaining both trustworthy identification and safe biometric embedding administration

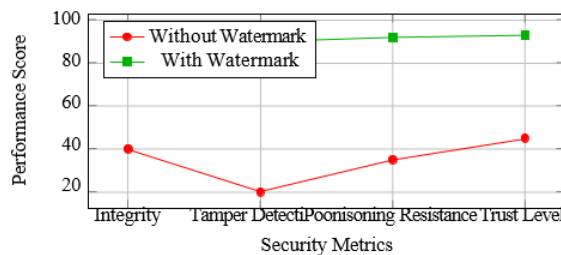


Figure 19: Security Performance Comparison

Fig. 19 reveals that the offered framework gives a high level of security with the negligible effect on the recognition accuracy. This is unlike the traditional methods that consider only recognition performance, DeepAudit incorporates embedding integrity checking and auditing of data enhancing the credibility and dependability of the biometric systems [7]. The findings show that recognition is not compromised when secure embedding protection is used. Performance

Discussion

This part of the paper examines the trade-off between security and recognition performance, practical deployment, and system scalability. The discussion reveals the viability of the planned DeepAudit system architecture of biometric uses in the real world. The analysis takes into account the operational constraints, system robustness and performance attributes that are typically considered within secure biometric systems and machine learning based authentication systems [5].

Security vs Performance Trade-Off

The suggested watermark- based protection increases the embedding integrity at the expense of the security strength and recognition performance. Embedding-level watermarking alters the feature representations to add integrity information that can have a minor impact on the similarity score used in identity matching [6]. Hiking the watermark strength enhances the ability to detect tampering and adversarial vulnerability and can also create. slight impairment in the accuracy of recognition

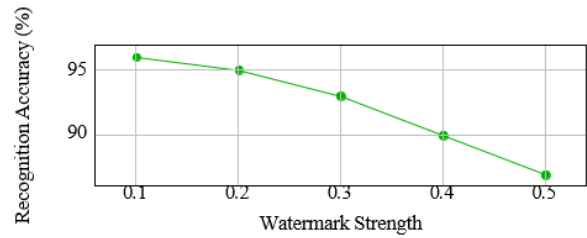


Figure 20: Security vs Performance Trade-Off

As Fig. 20 illustrates, the best watermark strength offers good integrity protection but has a low recognition accuracy

Practical Deployment Considerations

Implementation of secure face recognition systems should be based on operational constraints like computational overhead, storage needs and system maintenance. Security mechanisms used in practical biometric systems need to be light and efficient as needed to accommodate real-time processing and scale to large-scale applications [7].

Factor	Consideration
Computational Cost	Additional watermark processing overhead
Storage Requirement	Secure storage for embeddings
System Maintenance	Periodic integrity verification
Security Management	Access control policies
Real-time Pro- cessing	Efficient embedding comparison

Table 10: Practical Deployment Considerations

Table 10 is the summary of practical implementation of deploying the DeepAudit watermarking framework into face recognition systems. The watermark embedding and verification procedure adds extra computations but the overhead is moderate as embedding operations are simple and feature-level integration is simple [8]. Watermarked embeddings would need additional memory and access controls to be secured to ensure that it is not changed unauthoriz- edly. All of this suggests that the suggested framework can be integrated with the current recognition systems with a reasonable amount of overhead and can also improve the data integrity and security [9].The design is useful in the applications like identity verificationsystems, access control, and surveillance.

Scalability Analysis

DeepAudit can be scaled by discussing the performance of the system when the dataset size is maximized. Similarity computation and computation of embeddings are said to require more computation as the number of stored

embeddings increases. integrity check also increase. This means that large-scale biometric systems require efficient embedding comparison and database management [10].

Number of Stored Embeddings

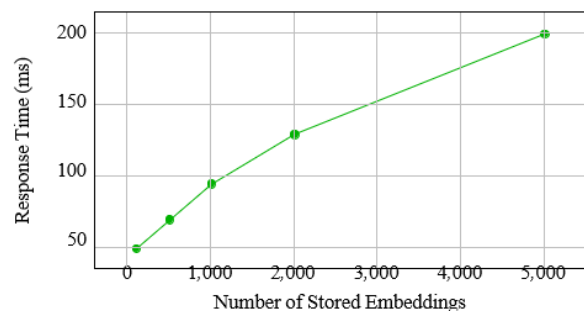


Figure 21: Scalability Analysis of DeepAudit Framework

Figure 21: Scalability Analysis of DeepAudit Framework

Fig. 21 results reveal that the response time increases slowly with the number of stored embeddings, which indi- cates

that the DeepAudit model is capable of scaling with moderate computational demands. The embedding based matching approach can be efficiently used to verify identity when large amounts of data are involved, whereas watermark verification has slight overhead [10]. Such results prove it is possible to implement DeepAudit in large-scale biometric system of reasonable performance

Limitations

Although the suggested DeepAudit framework is effective to assure the implementation of integrity and secure recognition, various constraints can be identified. These drawbacks are based on the nature of biometric recognition systems environmental parameters that influence the performance of the system, and constraints in data quality and model generalization. The same limitations have been cited in the existing face recognition and biometric authentication studies indicating difficulties in attaining quite robust and universal-reliable biometric systems [11].

Biometric Limitations

Face recognition systems have the ability to identify by the distinctive features of the face. Nevertheless, biometric systems are always associated with the problems of high similarity between the individuals, sensitivity to the changes in the faces, and within-class alterations that occur due to aging and changes in expression [1]. Such restrictions can influence the reliability of recognition in some situations, especially in unstructured ones.

These issues are prevalent to the vast majority of face recognition systems and cannot be entirely handled even by the protection mechanisms implemented in the embedding level. Even though deep learning models enhance resistance to fluctuations, recognition is enhanced can continue to degenerate even in extreme conditions or in situations where there is scanty training data [2]. More so, the biometric uniqueness assumptions cannot be always true in the real world which can create an ambiguity in classification.

Limitation	Impact
Identical Twins	High facial similarity causes misclassification
Aging Effects	Facial changes reduce recognition accuracy
Facial Expressions	Variations affect feature extraction
Pose Variation	Non-frontal views reduce matching reliability
Occlusion	Masks or glasses obstruct facial features

Table 11: Biometric System Limitations

Table 11 is a general overview of the pitfalls with biometric systems, especially facial recognition biometrics. Similarity in identical twins might result in misclassification of

similarity and aging effects might decline testing accuracy over time. Facial expression differences, pose, and occlusions like masks or glasses may hinder the extraction of features and also reduce matching reliability. These limitations should be known in order to design strong recognition systems and to add other security measures like watermark based embedding integrity checks.

Environmental Constraints

The lighting, image quality, background complexity and sensor capabilities, are environmental factors that greatly affect the face detection and feature extraction performance. These forces can worsen recognition. verify and accuracy, especially in a real-time deployment environment [3].

Table 12: Environmental Constraints

Constraint	Effect
Poor Lighting	Reduces feature visibility
Low Image Resolution	Loss of discriminative features
Background Noise	Detection errors
Camera Quality	Affects image clarity
Motion Blur	Distorts facial details

Table 12 is a summary of typical factors in the environment that may affect the performance of facial recognition systems. The inadequate light conditions decrease the visibility of features, whereas the low quality of the images and

the camera may lead to discriminatory purgation. Motion blur and background noise also add to error of detection and distortion of the features of the face. These environmental factors must be taken into consideration during the design of powerful systems and in the process of implementation further protection like watermark based embedding integrity validation.

These limitations can be overcome through better preprocessing methods, stronger feature extraction models and controlled acquisition environments. Although the DeepAudit framework increases the incorporation of integrity, it does not causally reduce environmental differences to feature extraction quality. Thus, the performance of a system is still subject to the quality of input images and conditions of their acquisition [4].

Moreover, there are other constraints that are brought about at the system level by factors like computational capacity, and the issue of scalability. Watermark embedding incurred only a small overhead cost, but it may necessitate large-scale implementation database management optimized and effective matching algorithms to ensure real-time performance [5]. These issues are an opportunity to conduct research on secure and scalable biometric systems in the future [6].

Conclusion and Future Work

This part is a summary of the main contributions of the proposed DeepAudit framework and a description of the possible directions of further research and system

improvement. The given work touches upon the security and integrity issues. in face recognition systems through embedding protection with biometric recognition, thus enhancing system trustworthiness, reliability, and adverse manipulation.

Summary of Contributions

This publication introduced DeepAudit, a safe face recognition model to assurance that the integrity and the dependability of face embeddings in biometric systems. In contrast to a traditional recognition system, the proposed framework will concentrate on training data protection, the built-in integrity verification, and the ability to resist adversarial manipulation, contrary to the traditional recognition system that mostly focuses on accuracy and optimization of the performance. The suggested system presents a safe face recognition pipeline combining watermark-based embedding protection to increase the integrity of the biometric data as well as avoid unauthorized alteration of stored embeddings.

A watermarking embedded mechanism on the embedding-level is created to allow integrity checks and tampering in the stored facial representations without impairing recognition performance. The work also gives a mathematical development of recognition and attack models that can analyze the system behavior and assess vulnerabilities in biometric systems. Secure enrollment, recognition and verification processes are conducted on the basis of a modular architecture to provide scalability and maintainability of the system. Experimental assessment has proven that the proposed method has high recognition accuracy and low effects of watermark embedding besides enhancing robustness in case of data poisoning and unauthorized embedding change. The results have demonstrated the need to include security measures in machine learning-based biometric systems and are a step in the right direction of creating such systems secure authentication systems of physical software.

Future Work

In spite of the fact the proposed framework offers effective embedding protection and integrity verification, there are a number of im- Future research can be used to explore improvements that can be made to the system to improve its performance, scalability, and security. Future there can be the addition of multi-factor authentication of a product to the work to enhance identity verification and im- provide reliability of systems in applications with security consequences. Detection Advanced anomaly-based techniques of poisoning.

Adversarial resistance can also be increased by creating detection or machine learning-based defense strategies. attacks. The suggested watermarking system can be pre-empted to other forms of biometrics like fingerprints and iris identification systems to offer holistic biometrics security. Also, watermark robustness optimization resistant

to advanced adversarial and backdoor attacks is a fruitful research area. Future research may also aim at serving big-scale deployments by using distributed secure storage and effective embedding management techniques. Moreover the implementation of blockchain-based logging tools to have secure audit trails and biometric authentication systems can be made more transparent and responsible with the help of tamper-proof record management.

Such advances can further enhance the security, scalability and reliability of biometric recognition systems in real-world environments

REFERENCE

- [1] J. Deng, J. Guo, J. Yang, N. Xue, I. Kotsia, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2019.
- [2] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2015.
- [3] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2014.
- [4] M. Turk and A. Pentland, "Eigenfaces for Recognition," Journal of Cognitive Neuroscience, vol. 3, no. 1, pp. 71–86, 1991.
- [5] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 711–720, 1997.
- [6] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.
- [7] B. Biggio, B. Nelson, and P. Laskov, "Poisoning Attacks Against Support Vector Machines," Proc. International Conference on Machine Learning (ICML), 2012.
- [8] T. Gu, B. Dolan-Gavitt, and S. Garg, "BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain," IEEE Access, vol. 6, pp. 47230–47244, 2018.
- [9] G. Xia, J. Chen, C. Yu, and J. Ma, "Poisoning Attacks in Federated Learning: A Survey," IEEE Access, vol. 11, pp. 10708–10730, 2023.
- [10] J. Terven, D.-M. Cordova-Esparza, J.-A. Romero-González, A. Ramírez-Pedraza, and E. A. Chávez-Urbiola, "A Comprehensive Survey of Loss Functions and Metrics in Deep Learning," Artificial Intelligence Review, vol. 58, 2025.
- [11] Yusuke Uchida, Yuki Nagai, Shigeyuki Sakazawa, and Shin'ichi Satoh, "Embedding Watermarks into Deep Neural Networks," Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval, pp. 269–277, 2017