American Journal of Applied Research and AI (AJARAI)

VOLUME 1 ISSUE 2 (2026)

PUBLISHED BY
E-PALLI PUBLISHERS, DELAWARE, USA

American Journal of
Applied Research and AI (AJARAI))

Volume 1 Issue 2, Year 2026
ISSN: 3142-8932 (Online)
DOI: https://doi.org/10.54536/ajarai.v1i2.7972
https://journals.e-palli.com/home/index.php/ajarai

Deep Neural Networks for Enhancing Robustness in Facial Authentication and Deepfake Detection for Driven Data Poisoning Attacks and Defense Strategies

Anas A. Nicola[1*]

Article Information

Received: April 02, 2026
Accepted: June 08, 2026
Published: September 09, 2026

Keywords

Algorithms, Deep Learning, Deep Neural Embedded, Intelligent System, Neural Network

ABSTRACT

This research combines two crucial works aimed at enhancing the robustness of deep neural networks (DNNs) against emerging attacks. In a model or training data extraction attack, an attacker analyzes the input, output, and other external information of a system to speculate on the parameters or training data of the model. Similar to the concept of Software-as-a-Service (SaaS) proposed by cloud service providers. These services are open, and users can use open APIs to perform image and voice recognition. In addition, facial authentication, altered to depict faces that were not originally present. In response, a new defense mechanism named Sanitization and Noise defenses (Deep- neural-network and Embedded Feature-based detector) is introduced. Sanitization and Noise defenses utilize a hybrid DNN and KNN (k-nearest neighbors) model to detect contaminated feature vectors generated by the attacked system. Evaluations on real datasets demonstrate Sanitization and Noise defenses efficacy, achieving over 80% detection accuracy across diverse settings. A novel data poisoning attack scenario is proposed where an attacker injects a few photos during a user's registration or photo update process. The proposed attack scenario involves an attacker injecting their photos into the facial recognition system during user registration or photo updates.

INTRODUCTION

The reliability associated with modern artificial intelligence (AI) models plays an important role in a wide range of applications, including Internet-of-Things. Consequently, over the last couple of years, these machine learning (ML) models demand adopting additional techniques in order to address security related issues since newly emerging vulnerabilities are being discovered and capable of posing a threat to the integrity of the ML model which is the target of an attacker(Ramirez *et al.*, 2022). Facial authentication is increasingly prevalent for unlocking personal devices like smartphones and laptops. The next frontier for this technology is its integration into web services According to recent statistics (BiggioNelson and Laskov, 2012), an average internet user manages 26 online accounts with only 5 unique passwords, highlighting the challenge of password management. Facial authentication offers a convenient solution, enabling users to log in simply by facing their device's camera, accessible across multiple devices without prior setup. This convenience has spurred the development of facial recognition APIs(Jebreel *et al.*, 2024) , poised for widespread adoption in web services. However, the security of facial authentication systems is paramount. These systems rely on cutting-edge techniques like FaceNet , built on DNNs (SuVargas and Sakurai, 2019), which, despite their accuracy, are susceptible to various attacks and model stealing (Moosavi-Dezfooli *et al.*, 2017). Adversarial and data poisoning attacks pose significant threats to facial authentication, potentially leading to misclassification of users or unauthorized access. A novel data poisoning attack scenario is proposed, where an attacker injects a few photos during a user's registration or photo update process, leading the system to classify both the genuine user and the attacker as the same person. This attack exploits vulnerabilities in home networks and routers (GoodfellowShlens and Szegedy, 2014), polluting the training dataset and compromising system integrity. Unlike traditional data poisoning attacks, this method doesn't require insider information or server compromise, making it stealthy and effective at granting unauthorized access. In response, a new defense mechanism named Sanitization and Noise defenses (Deep- neural-network and Embedded -based detector) is introduced. Sanitization and Noise defenses utilize a hybrid DNN and KNN (k-nearest neighbors) model to detect contaminated feature vectors generated by the attacked system. Evaluations on real datasets demonstrate Sanitization and Noise defenses efficacy, achieving over 90% detection accuracy across diverse settings. Traditional trust in photographic evidence is eroded by deepfake technology, which leverages advanced AI techniques like Generative Adversarial Networks (GANs) (Metzen *et al.*, 2017). Existing detection methods, achieve high accuracy but are vulnerable to data poisoning attacks during training (XuEvans and Qi, 2017). Unlike adversarial input attacks, data poisoning in deepfake detection involves manipulating training data labels rather than perturbing the images themselves, making it challenging to detect using traditional defenses. To mitigate this, a novel defense mechanism is proposed the Protector

[1] Faculty of Telecommunication, Engineering and Space Technology, Future University, Khartoum, Republic of the Sudan
[*] Corresponding author's email: anasnicola2018@gmail.com

Network. This additional neural network is appended to existing deepfake detectors, specifically trained to identify poisoned data labels. Unlike the detector itself, the Protector Network's classification remains robust against evolving types of fake images, ensuring sustained effectiveness.

LITRETURE REVEW

Feedforward Neural Networks (FNNs),Present one of the simplest and most fundamental architectures in neural networks, where data flows in a unidirectional manner from input to output without any cycles or loops. Each node in a layer connects fully to every node in the subsequent layer, creating a fully connected structure. The research will provide a comprehensive overview of the results , architecture consists of several components: the input layer, where data enters the model, with each neuron corresponding to a specific feature such as pixel values in an image; hidden layers, which lie between the input and output layers and are responsible for extracting and learning features through linear transformations followed by non-linear activation functions (CohenSapiro and Giryes, 2020); and the output layer, which produces the final output, such as a probability distribution for classification or continuous values for regression. Study (Meng and Chen, 2017), introduce non-linearity, activation functions like ReLU (Rectified Linear Unit) are often employed. ReLU outputs zero for negative values and the input itself for positive values, making it computationally efficient and reducing the likelihood of vanishing gradients. In classification tasks, the soft max function is typically used in the output layer to convert logits into probabilities, ensuring that the sum of the probabilities equals 1, which is essential for multi-class problems summarized in (Nelson *et al.*, 2008). The training process involves two main steps: forward propagation and backpropagation. In forward propagation, input data passes through the network, and activations are calculated layer by layer to generate predictions (Lowd and Meek, 2005). However, in their research (RumelhartHinton and Williams, 1986). Explained backpropagation follows by computing the error between the predicted output and the target using a loss function, and the weights are updated to minimize this loss. A common regularization technique used to prevent overfitting is dropout, where a fraction of neurons is randomly ignored during each training iteration, creating a more robust model by preventing co-adaptation of neurons detailed in the research (Wittel and Wu, 2004). During the training phase, a specified dropout rate, such as 0.5, sets 50% of the neurons to zero, while the remaining neurons are scaled up to maintain the expected activations (Kalyani and Akhila, 2025). At the testing phase, dropout is turned off, and the full network is used with appropriately scaled weights. This technique improves generalization by forcing the network to learn independent features, making it more robust to noise and variations (Wang and Gong, 2018). Dropout is commonly applied to fully connected layers, with typical dropout rates ranging from 0.2 to 0.5, depending on the model complexity and training data (OrekondySchiele and Fritz, 2019) .

Deep neural networks DNNs

deep neural networks (DNNs) in the domains of facial authentication software and deepfake detection. Facial authentication offers a convenient solution, enabling users to log in simply by facing their device's camera, accessible across multiple devices without prior setup. This convenience has spurred the development of facial recognition APIs (Lee *et al.*, 2019), poised for widespread adoption in web services.

DNN-Based Facial Authentication Model

The security of facial authentication systems is paramount, (Juuti *et al.*, 2019). These systems rely on cutting-edge techniques like FaceNet, built on DNNs (Wang and Gong, 2018), which, despite their accuracy, are susceptible to various attacks including adversarial inputs data poisoning, and model stealing Adversarial and data poisoning attacks pose significant threats to facial authentication, potentially leading to misclassification of users or unauthorized access (Paudice *et al.*, 2018). A novel data poisoning attack scenario is proposed (Figure 1), where an attacker injects a few photos during a user's registration or photo update process, leading the system to classify both the genuine user and the attacker as the same person. This attack exploits vulnerabilities in home networks and routers (Goswami *et al.*, 2018),polluting the training dataset and compromising system integrity(Aithal *et al.*, 2024) .Unlike traditional data poisoning attacks, this method doesn't require insider information or server compromise, making it stealthy and effective at granting unauthorized access.

Label Manipulation

Attackers maliciously alter the labels of selected data points by flipping them from their true classes to incorrect ones. The manipulated data is then injected back into the training dataset, along with legitimate data, to train the model.

Training Phase

The model is trained on the manipulated dataset, where a portion of the data contains flipped labels. As a result, the model may learn incorrect associations between features and labels, leading to compromised performance and erroneous predictions.

Consequences

Label flipping attacks can cause the model to misclassify future instances, leading to potentially harmful decisions in real-world scenarios. The impact of these attacks can be severe, especially in critical applications such as healthcare, finance, and autonomous systems.

Rise of Facial Authentication Model

With facial authentication in place, internet users simply position themselves in front of the device's camera to log into web services. This is convenient, swift, and accessible, being able to be done from a multitude of devices without additional preparations. Its promising market potential has fostered several releases of facial recognition APIs. It is envisioned that facial authentication will be widely adopted in web services in the near future.

DNN-Based Facial Authentication Model

Rise of Facial Authentication

Facial authentication technology has become increasingly prevalent for unlocking personal devices. Therefore, Facial authentication simplifies this process by allowing users to access web services through a camera, eliminating the need to manually manage passwords. The convenience and accessibility of this technology have driven the development of numerous facial recognition APIs, signaling a promising market expansion see Figure 1.

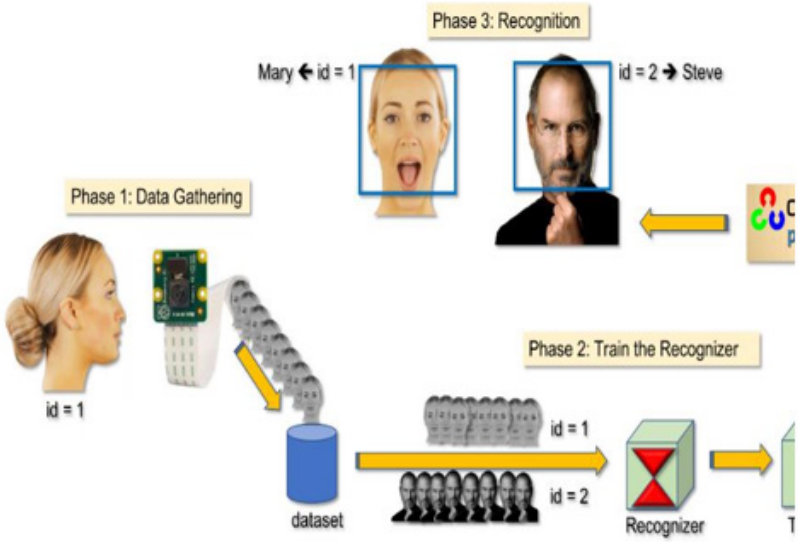

Figure 1: Face recognition Process

Security Challenges in Facial Authentication

Modern facial recognition techniques, such as FaceNet, rely on DNNs that are vulnerable to various attacks, including adversarial input attacks and data poisoning attacks. In the context of web services, these attacks can lead to serious security breaches, such as misclassifying legitimate users or granting unauthorized access to attackers. Data poisoning attacks, in particular, pose a severe threat by contaminating the training data of facial recognition systems see Figure 2.

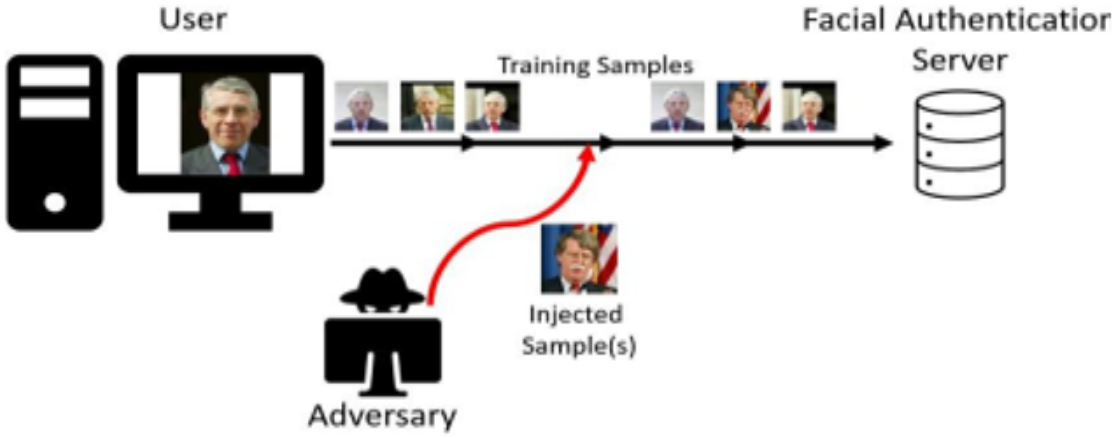

Figure 2: Proposed Data Poisoning Attack

Attacks on Deep Neural Networks (DNNs)

Understanding the landscape of attacks on DNNs is essential for developing effective defenses. Adversarial input attacks and data poisoning attacks are two primary methods used to compromise DNNs. Adversarial attacks introduce perturbations to input data to mislead classifiers, while data poisoning attacks manipulate training data to alter the model's behavior.

Adversarial Attacks on Deep Neural Networks

Adversarial attacks aim to deceive DNNs by introducing subtle perturbations to input data. These perturbations are often imperceptible but can cause significant misclassifications. Various methods for generating adversarial examples include:

Gradient-Based Methods: These involve computing perturbations based on the model's loss function gradient (Seals, 2020).

One-Pixel Attacks: These alter a minimal number of pixels to induce misclassifications.

Boundary-Based Attacks: These exploit the decision boundaries of the target model.

Universal Adversarial Perturbations: These create perturbations applicable to multiple images.

Adversarial Image Generation: These involve generating entirely new adversarial images. Defenses against adversarial attacks include adversarial training, which enhances model robustness by incorporating adversarial examples into the training dataset (Pascu, 2020). attacks on facial recognition systems, the attackers must be careful to avoid altering the final classification accuracy in any significant way, which would cause alarm, potentially tipping off moderators that something is amiss (Eykholt *et al.*, 2018) . Thus, the goal of this attack is to maximize the probability that Image will be misclassified, while avoiding preventing Image and Image from being labeled correctly (ThiesZollhöfer and Nießner, 2019). The goal of attacking Images is formalized as G(Img, L), shown in Equation 1,2 and 3

$$G(\mathrm{Img}, \mathrm{L}) = \min_{\mathrm{n}} = \frac{l(Imgx,Lt)}{n} + \frac{l(Imgn,Lg)x^2}{s-n} + lmg(n+1) \ldots (1)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \ldots (2)$$

Definition The overall accuracy: evaluates the detection correctness for both the infected photos and pristine photos .Definition Recall: measures the percentage of the correctly identified infected photos against the total number of infected photos that have been tested (Aithal *et al.*, 2024). In the following equation, TP stands for "true positive" and FN stands for "false negative".

$$\text{Recall} = \frac{CorrectlyIdentifiedinfected}{ALLInfectedphoto} photo - \frac{\mathrm{TP}}{TP+FN} \ldots (3)$$

MATERIALS AND METHODS

Experiment setup

Datasets

The experiments utilize the CASIA-WebFace and CMU Multi-PIE datasets, which are widely used in facial recognition research. CASIA-WebFace Dataset, this dataset contains over 5,000 identities and is primarily used for training large-scale face verification systems. The images in CASIA-WebFace, offer a wide range of facial variations, making it ideal for training deep learning models to recognize faces in various real-world scenarios (Schwartz, 2018) . Its scale and diversity provide a solid foundation for evaluating the robustness of facial recognition systems (Escalante, 2020). CMU Multi-PIE Dataset: This dataset comprises images of 227 individuals captured under controlled conditions, with variations in pose, lighting, and facial expression. The CMU Multi-PIE dataset is particularly useful for testing models under different conditions and evaluating how well they generalize to new variations in appearance (Rossler *et al.*, 2019).

Experimental Design

The experimental design for this research aims to evaluate the impact of targeted data poisoning attacks on deep neural network (DNN)-based facial recognition systems, specifically leveraging the CASIA-WebFace and CMU Multi-PIE datasets. Additionally, it seeks to assess the effectiveness of various defense mechanisms in mitigating these attacks. The design includes several critical steps, each described in detail below:

Splitting Methodology

The datasets are split into training and testing sets following an 80/20 ratio, commonly used in machine learning experiments. The training set contains 80% of the total data, while the testing set comprises the remaining 20%.

Evaluation Metrics

The performance of the models is assessed using a suite of evaluation metrics that provide a comprehensive understanding of their resilience to data poisoning (Chollet, 2017). These metrics include:

Precision: Precision measures the proportion of correctly identified faces out of all the faces that the model classified as belonging to a certain identity. It helps quantify the model's accuracy when recognizing poisoned data, particularly the level of false positives

Recall: Recall measures the proportion of actual positive faces (correct identities) that the model successfully identifies. In the context of poisoning, this metric reveals how much the poisoning attack impacts the model's ability to correctly classify true identities.

F1 Score: The F1 score is the harmonic mean of precision and recall, providing a balanced measure of the model's overall performance. It is particularly useful when comparing models trained on clean versus poisoned data, as it highlights the trade-off between false positives and false negatives in facial recognition tasks.

Accuracy: Accuracy measures the overall percentage of correct classifications out of all predictions. This metric serves as a baseline for comparing the performance of models on clean and poisoned datasets and is a standard measure for facial recognition systems.

RESULTS AND DISCUSSION

Impact of Data Poisoning

Data poisoning attacks can significantly affect the performance of deep neural networks (DNNs) used in facial recognition systems. The experiments conducted

reveal crucial insights into how different poisoning strategies impact model metrics and the effectiveness of various defense mechanisms see Figure 3.

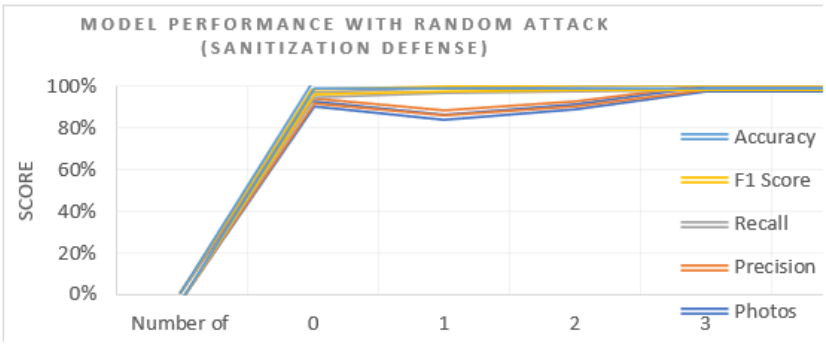

Figure 3: Accuracy of the model with and without data poisoning

Figure 3, explain the performance with random attack. The first set of results illustrates the model's performance across different metrics Precision, Recall, F1 Score, and Accuracy—while increasing the number of poisoned photos. Initially, Precision peaks with around 10 poisoned photos but sharply declines as the number of adversarial examples grows. Recall remains low throughout, suggesting persistent challenges in identifying true positives. Similar issues have been noted where adversarial attacks hindered the model's ability to generalize, resulting in low Recall. The F1 Score also remains consistently low, reflecting an imbalance between Precision and Recall, a phenomenon previously reported, where poisoning attacks disrupted performance metrics. Accuracy, while relatively stable, may mask underlying vulnerabilities. It has been emphasized that accuracy alone does not capture the full impact of poisoning, as it overlooks the Precision-Recall trade-offs crucial for assessing model robustness. See Figure 4 (a), and (b). Analysis performance Attack without noise defense Figure 5, shown Training Matrices random attack with sanitization attack photos accuracy score. Therefore, Figure 6, shown the number of injected photos with accuracy of the model with and without data poisoning.

	Name	Distribution	Mean	Mode	Median	Dispersion	Min.	Max.	Missing
N	Photos		17.40	3	12	0.73	3	35	1 (17 %)
N	Precision		0.117793	0.05126	0.08926	0.514658	0.05126	0.227975	1 (17 %)
N	Recall		0.42320	0.080	0.094	0.96893	0.080	0.950	1 (17 %)
N	F1 Score		0.0536928	0.049818	0.05544	0.0601183	0.049818	0.0578	1 (17 %)
N	Accuracy		0.09580	0.088	0.095	0.06101	0.088	0.105	1 (17 %)

(a)

	Name	Distribution	Mean	Mode	Median	Dispersion	Min.	Max.	Missing
N	Photos		17.40	3	12	0.73	3	35	1 (17 %)
C	Feature 3			20		1.24			0 (0 %)
C	Number of			0		1.79			0 (0 %)

(b)

Figure 4: Analysis Performance with Random Attack (Noise Defense)

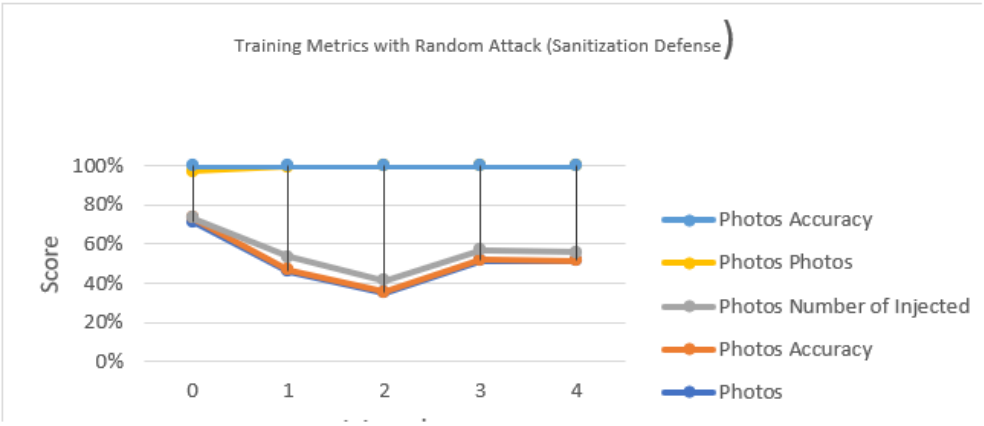

Figure 5: Training Matrices random attack with sanitization attack

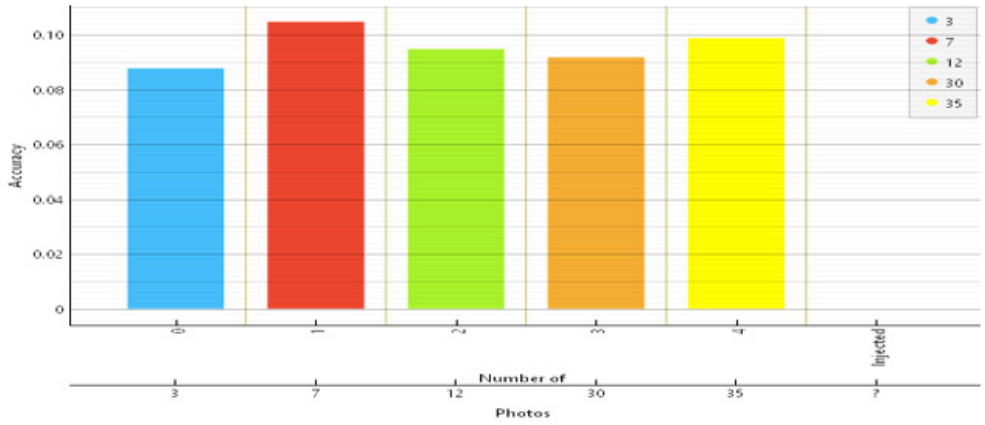

Figure 6: Accuracy of the model with and without data poisoning.

Training Metrics with Random Attack (No Defense). A detailed table 1, breakdown of the model's performance across different levels of poisoning (1, 5, 10, and 20 injected photos) reveals several trends. Precision initially increases with more poisoned photos up to 10, indicating temporary confidence. However, this is followed by a sharp drop as the number of poisoned photos rises, consistent with finding.

Table 1: Results Table for Random Attack (No Defense)

Number of Injected	Photos	Precision	Recall	F1 Score	Accuracy
0	3.0	0.051260	0.080	0.049946	0.088
1	7.0	0.131220	0.90	0.049818	0.105
2	12.0	0.227975	0.95	0.057800	0.095
3	30.0	0.089250	0.092	0.055440	0.092
4	35.0	0.089260	0.094	0.055460	0.099

A detailed breakdown of the model's performance across different levels of poisoning (3 ,7 ,12, 30 and 35 injected photos) reveals several trends. Precision initially increases with more poisoned photos up to 10, indicating temporary confidence. However, this is followed by a sharp drop as the number of poisoned photos rises, consistent with findings. See figure 5, table 2, Random Attack (Noise Defense), figure 5, explained details number of injected photos in the model training 0.09,0.10,0.085. Figure 6, shown the curve of the number attack simulated without defense the total injected photo between 0.4- 5 shown in the table 2, full details.

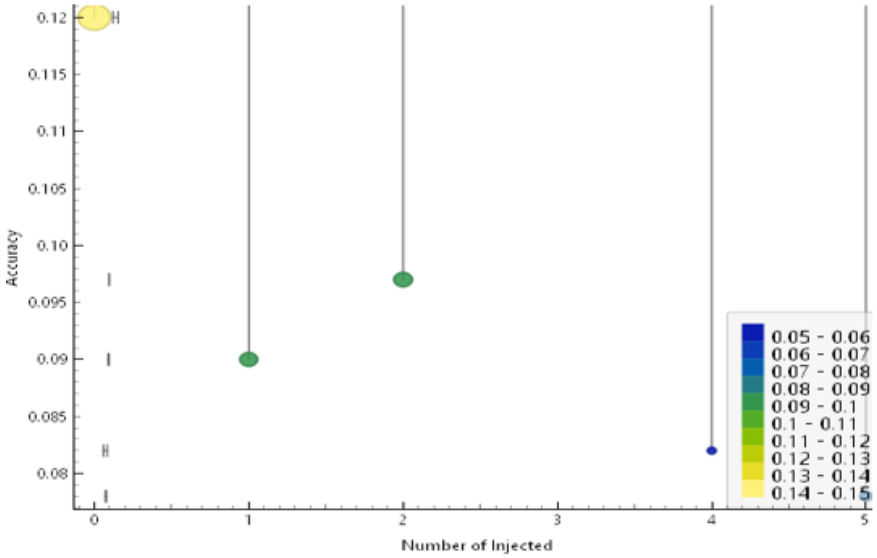

Figure 5: Training Metrics with Random Attack (No Defense)

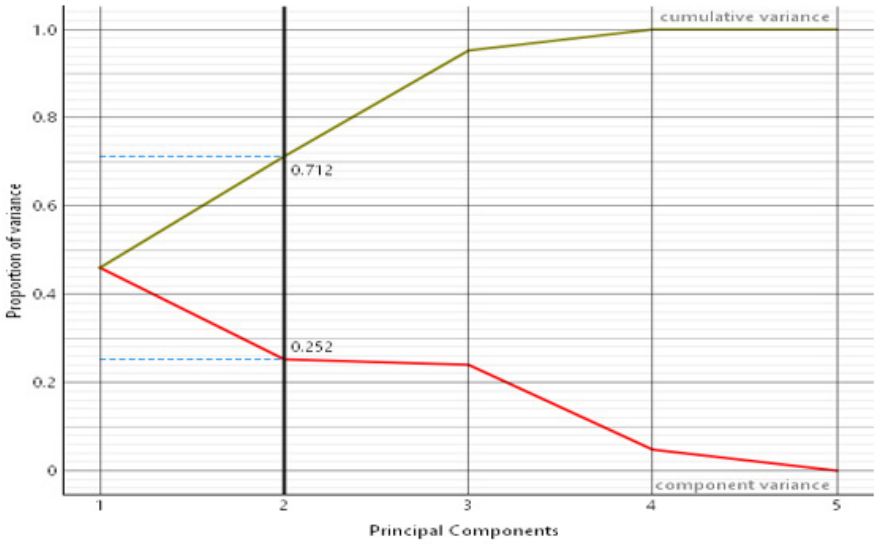

Figure 6: Cursive Metrics with Random Attack with (No Defense)

Table 2: Results Table for Random Attack (Noise Defense)

Number of Injected	Photos	Precision	Recall	F1 Score	Accuracy
0	1.0	0.153027	0.120	0.055668	0.098
1	7.0	0.094857	0.090	0.056668	0.099
2	20.0	0.095398	0.096	0.094788	0.097
3	25.0	0.057744	0.088	0.061791	0.082
4	30.0	0.071751	0.078	0.058034	0.078

Model Performance with Random Attack (Sanitization Defense)

Graph Figure 6, presents the model's performance under random poisoning attacks, with a sanitization defense applied. Four metrics—Precision, Recall, F1 Score, and Accuracy— are evaluated as the number of injected (poisoned) photos increases from 1 to 20. Here's a breakdown of each metric and the implications of this figure: Figure 7, illustrates the model's performance under random attacks, with a sanitization defense in place. While the defense offers some resilience against smaller attacks, its effectiveness declines as the poisoning becomes more

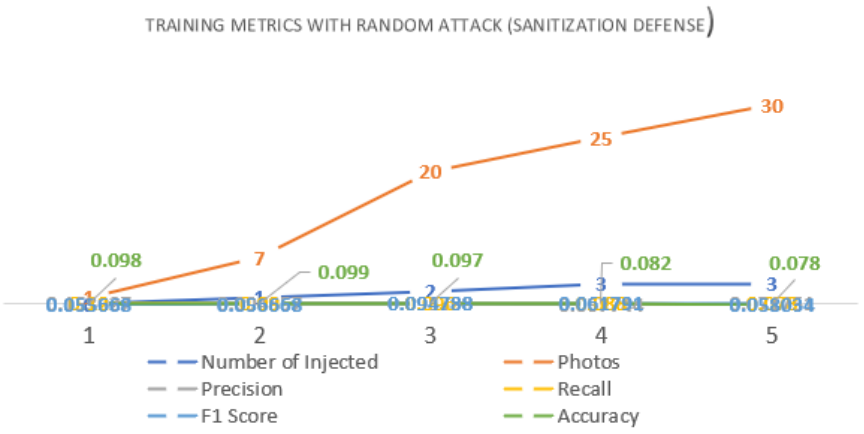

Figure 6: Model Performance with Random Attack (Noise Defense)

aggressive. The details of number of total injected photos is shown in the table figure 8, gives numbers of sorting injected photos for Random Attack (Noise Defense).

- Recall: Recall measures the model's ability to identify all true positives. Starting from +0.900 with no poisoned photos, recall decreases as more photos are injected.

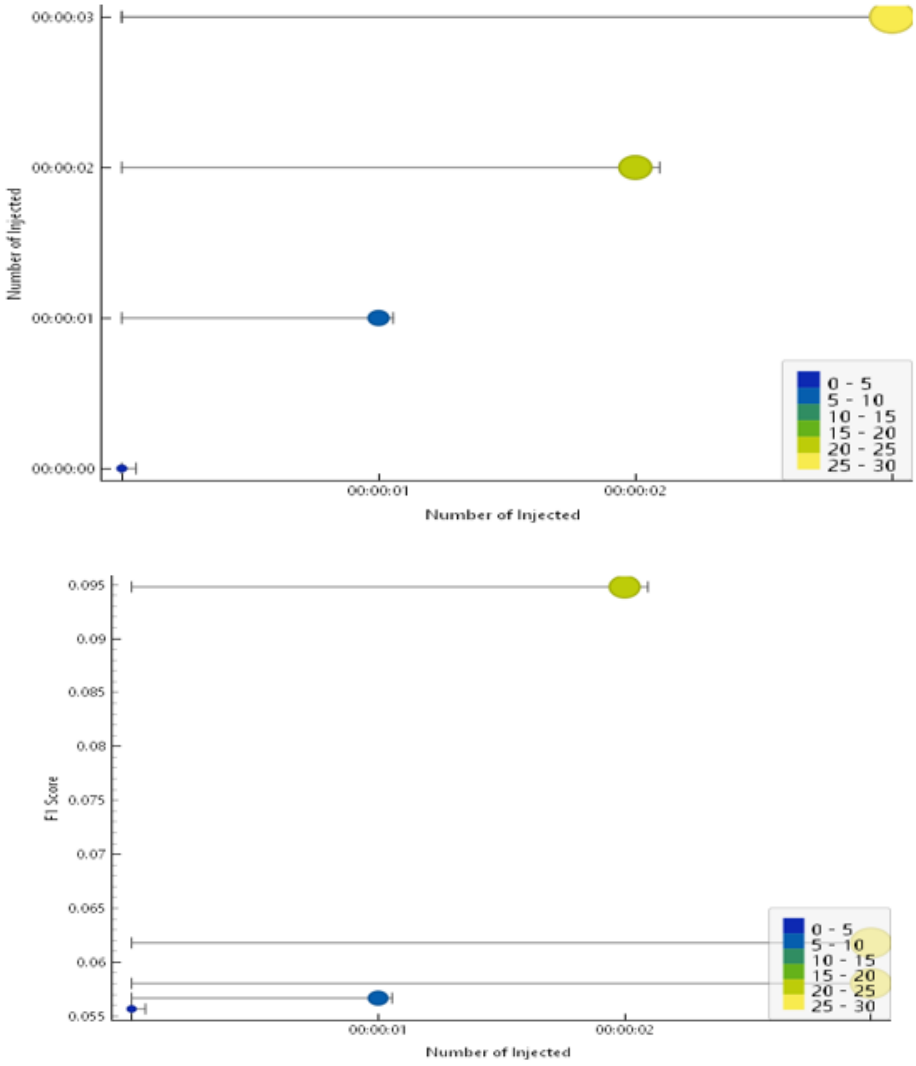

Figure 7: Number of injected photos

1	+0.975	Number of Injected	:	Photos
2	-0.900	Photos	:	Recall
3	-0.900	Accuracy	:	Photos
4	+0.900	Precision	:	Recall
5	-0.872	Number of Injected	:	Recall
6	-0.872	Number of Injected	:	Precision
7	-0.872	Accuracy	:	Number of Injected
8	-0.800	Photos	:	Precision
9	+0.700	Accuracy	:	Recall
10	+0.667	F1 Score	:	Number of Injected
11	+0.600	F1 Score	:	Photos
12	+0.600	Accuracy	:	Precision
13	-0.500	Accuracy	:	F1 Score
14	-0.400	F1 Score	:	Precision
15	-0.300	F1 Score	:	Recall

Figure 8: Results Table for Random Attack (Noise Defense)

With 1 poisoned photo, recall is - 0.300, then - 0.800 with photos. The decrease in recall signifies that the model's capacity to detect true positives diminishes as the noise level increases.

F1 Score: The F1 Score, which balances precision and recall, follows a similar downward trend. It starts at + 0.500 with no poisoned data, and = 0.667 number of injected photos.

Accuracy: Accuracy measures the overall correctness of the model's predictions. Initially, accuracy is + 0.600 with no poisoned photos and decreases progressively to -0 .500 with photos accuracy F1 score.

The Result: The results underscore the importance of employing a multi-faceted approach to adversarial defense, combining various techniques to achieve more robust protection. The observed trends in precision, recall, F1 Score, and accuracy suggest that a singular defense strategy may be inadequate, and a hybrid approach could offer more effective protection against adversarial data.

Discussion

Future research could explore combining sanitization with other defense mechanisms to enhance the model's robustness against extensive adversarial attacks. The results table for the random attack with noise defense presents performance metrics of a model across varying levels of poisoned data. The metrics—Precision, Recall, F1 Score, and Accuracy—are evaluated at four different levels of injected photos. The defense mechanism provides some mitigation against poisoning but is limited by the scale of the attack. Its effectiveness diminishes as more adversarial data is introduced, suggesting that this defense strategy may be better suited for low- to moderate-level poisoning but less effective against larger attacks. The simulation result shown more than 70 % number of injected photos without noise defense presented in the table 1, 0. 3, 0.7, 0.12, 0.30 ,0.35, as percentage 0.0.95,0.09,0.08 with random attack without defense 0.05 shown in Training Metrics with Random Attack, figure 5. Table 2 figure 6. The number of injected photos 1.0, 7.0, 20.0, 25.0 ,30.0. The graph shown the number of injected photos: 0.056 – accuracy 0.099, 0.096 accuracy 0.099, 0.098 accuracy 0,097, 0.061 accuracy 0.082, 0.058 accuracy 0.078. Final report explained in the Table for Random Attack (Noise Defense). In table known Figure 8, the total summary of photo injected with no defense (- 0.872), with noise defense (+.667).

CONCLUSION

This research explored the vulnerabilities of DNN-based deepfake detectors to targeted data poisoning attacks and proposed a novel defense mechanism, the Protector network. Through extensive experiments, we demonstrated that even small amounts of poisoned data can severely compromise the performance of deepfake detectors. However, the Protector network successfully mitigates this threat by accurately identifying and filtering poisoned data, ensuring the reliability and security of deepfake detection systems. Comparative analysis with other defense mechanisms confirmed that the proposed model offers significant improvements in accuracy, robustness, and adaptability to evolving attack strategies. Data poisoning has a profound effect on the performance of facial recognition systems. Our experiments demonstrated that even a minimal amount of poisoned data can significantly degrade essential performance metrics, such as precision, recall, and F1 score Specifically.

REFRENCES

Aithal, S. K., Maini, P., Lipton, Z. C., & Kolter, J. Z. (2024). Understanding hallucinations in diffusion models through mode interpolation. *Advances in neural information processing systems, 37*, 134614-134644.

Biggio, B., Nelson, B., & Laskov, P. (2012). Poisoning attacks against support vector machines. In *Proceedings*

of the 29th International Conference on Machine Learning (ICML-12) (pp. 1467–1474).

Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1251–1258). IEEE. https://doi.org/10.1109/CVPR.2017.195

Cohen, G., Sapiro, G., & Giryes, R. (2020). Detecting adversarial samples using influence functions and nearest neighbors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 14453–14462). IEEE. https://doi.org/10.1109/CVPR42600.2020.01446

Escalante, A. (2020, August 3). Research finds social media users are more likely to believe fake news. Forbes. *Forbes Magazine, August, 3*.

Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., & Song, D. (2018). Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1625–1634). IEEE. https://doi.org/10.1109/CVPR.2018.00175

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*. https://doi.org/10.48550/arXiv.1412.6572

Goswami, G., Ratha, N., Agarwal, A., Singh, R., & Vatsa, M. (2018). Unravelling robustness of deep learning based face recognition against adversarial attacks. *Proceedings of the AAAI Conference on Artificial Intelligence, 32*(1), 6829–6836. https://doi.org/10.1609/aaai.v32i1.12341

Jebreel, N. M., Domingo-Ferrer, J., Sánchez, D., & Blanco-Justicia, A. (2024). LFighter: Defending against the label-flipping attack in federated learning. *Neural Networks, 170,* 111–126. https://doi.org/10.1016/j.neunet.2023.11.019

Juuti, M., Szyller, S., Marchal, S., & Asokan, N. (2019). PRADA: Protecting against DNN model stealing attacks. In *2019 IEEE European Symposium on Security and Privacy (EuroS&P)* (pp. 512–527). IEEE. https://doi.org/10.1109/EuroSP.2019.00044

Kalyani, K., & Akhila, G. (2025). Secure distributed learning: Robust data poisoning detection framework using intelligent model integrity analysis. *International Journal of Data Science and IoT Management System, 4*(4), 230–235. https://doi.org/10.64751

Lee, T., Edwards, B., Molloy, I., & Su, D. (2019). Defending against neural network model stealing attacks using deceptive perturbations. In *2019 IEEE Security and Privacy Workshops (SPW)* (pp. 43–49). IEEE. https://doi.org/10.1109/SPW.2019.00020

Lowd, D., & Meek, C. (2005). Adversarial learning. *In Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 641–647). Association for Computing Machinery. https://doi.org/10.1145/1081870.1081950

Meng, D., & Chen, H. (2017). MagNet: A two-pronged defense against adversarial examples. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security* (pp. 135–147). Association for Computing Machinery. https://doi.org/10.1145/3133956.3134057

Metzen, J. H., Genewein, T., Fischer, V., & Bischoff, B. (2017). *On detecting adversarial perturbations.* arXiv. https://doi.org/10.48550/arXiv.1702.04267

Moosavi-Dezfooli, S.-M., Fawzi, A., Fawzi, O., & Frossard, P. (2017). Universal adversarial perturbations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1765–1773). IEEE. https://doi.org/10.1109/CVPR.2017.17

Nelson, B., Barreno, M., Chi, F. J., Joseph, A. D., Rubinstein, B. I. P., Saini, U., Sutton, C., Tygar, J. D., & Xia, K. (2008). Exploiting machine learning to subvert your spam filter. In *Proceedings of the 1st USENIX Workshop on Large-Scale Exploits and Emergent Threats* (pp. 7:1–7:9). USENIX Association.

Orekondy, T., Schiele, B., & Fritz, M. (2019). Knockoff nets: Stealing functionality of black-box models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 4954–4963). IEEE. https://doi.org/10.1109/CVPR.2019.00509

Pascu, L. (2020, June 29). Acronis reports critical flaws in GeoVision biometric devices, man-in-the-middle attack risks. *Biometric Update*

Paudice, A., Muñoz-González, L., György, A., & Lupu, E. C. (2018). *Detection of adversarial training examples in poisoning attacks through anomaly detection.* arXiv. https://doi.org/10.48550/arXiv.1802.03041

Ramirez, M. A., Kim, S.-K., Al Hamadi, H., Damiani, E., Byon, Y.-J., Kim, T.-Y., Cho, C.-S., & Yeun, C. Y. (2022). *Poisoning attacks and defenses on artificial intelligence: A survey. arXiv.* https://doi.org/10.48550/arXiv.2202.10276

Rössler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019). FaceForensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (pp. 1–11). IEEE. https://doi.org/10.1109/ICCV.2019.00009

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature, 323*(6088), 533–536. https://doi.org/10.1038/323533a0

Schwartz, O. (2018, November 12). You thought fake news was bad? Deep fakes are where truth goes to die. *The Guardian*. Original Guardian article

Seals, T. (2020, July 23). ASUS home router bugs open consumers to snooping attacks. *Threatpost.*

Su, J., Vargas, D. V., & Sakurai, K. (2019). One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation, 23*(5), 828–841. https://doi.org/10.1109/TEVC.2019.2890858

Thies, J., Zollhöfer, M., & Nießner, M. (2019). Deferred neural rendering: Image synthesis using neural

textures. *ACM Transactions on Graphics, 38*(4), Article 66. https://doi.org/10.1145/3306346.3323035

Wang, B., & Gong, N. Z. (2018). Stealing hyperparameters in machine learning. In *2018 IEEE Symposium on Security and Privacy (SP)* (pp. 36–52). IEEE. https://doi.org/10.1109/SP.2018.00038

Wittel, G. L., & Wu, S. F. (2004). On attacking statistical spam filters. In P*roceedings of the First Conference on Email and Anti-Spam (CEAS 2004)*. CEAS.

Xu, W., Evans, D., & Qi, Y. (2017). Feature squeezing: Detecting adversarial examples in deep neural networks. *arXiv*. https://doi.org/10.48550/arXiv.1704.01155