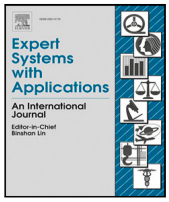
Expert Systems With Applications 250 (2024) 123941

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa



Audio-deepfake detection: Adversarial attacks and countermeasures

Mouna Rabhi [a,*], Spiridon Bakiras [b], Roberto Di Pietro [c]

[a] Division of Information and Computing Technology, College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar
[b] Infocomm Technology Cluster, Singapore Institute of Technology, Singapore
[c] RC3 Center, CEMSE Division, King Abdullah University of Science and Technology (KAUST), Saudi Arabia

ARTICLE INFO

Keywords:
Authentication
Adversarial attacks
Audio deepfake
Fake voice detection
GAN
Biometrics
Security

ABSTRACT

Audio has always been a powerful resource for biometric authentication: thus, numerous AI-based audio authentication systems (classifiers) have been proposed. While these classifiers are effective in identifying legitimate human-generated input their security, to the best of our knowledge, has not been explored thoroughly when confronted with advanced attacks that leverage AI-generated deepfake audio. This issue presents a serious concern regarding the security of these classifiers because, *e.g.*, samples generated using adversarial attacks might fool such classifiers, resulting in incorrect classification. In this study, we prove the point: we demonstrate that state-of-the-art audio deepfake classifiers are vulnerable to adversarial attacks. In particular, we design two adversarial attacks on a state-of-the-art audio-deepfake classifier, *i.e.*, the Deep4SNet classification model, which achieves 98.5% accuracy in detecting fake audio samples. The designed adversarial attacks[1] leverage a generative adversarial network architecture and reduce the detector's accuracy to nearly 0%. In particular, under graybox attack scenarios, we demonstrate that when starting from random noise, we can reduce the accuracy of the state-of-the-art detector from 98.5% to only 0.08%. To mitigate the effect of adversarial attacks on audio-deepfake detectors, we propose a highly generalizable, lightweight, simple, and effective add-on defense mechanism that can be implemented in any audio-deepfake detector. Finally, we discuss promising research directions.

1. Introduction

Sophisticated voice recognition systems have made human voice a relevant biometric tool for authentication. In addition, the generation of human-like voices has been studied for many decades. Owing to recent advancements in artificial intelligence (AI) technologies, the field of audio-based authentication is expanding—in terms of both proposed solutions and attacks against the proposed solutions. For example, many AI-enabled tools have been proposed to generate synthetic voices that are virtually indistinguishable from natural human voice and have been applied successfully in a wide range of fields, such as automated dubbing for movies, navigation systems, and speech assistants.

However, a significant security threat to voice biometrics arises when the cited technology is maliciously used. In particular, *audio-deepfake* technology involves the use of a trained AI model to generate realistic voice samples that are indistinguishable from the victim's real speech characteristics. In this context, researchers have proposed several models for synthetic audio generation, *e.g.*, (Engel et al., 2019; Gao, Singh, & Raj, 2018; Kumar et al., 2019; Oord et al., 2018, 2016; Wang et al., 2017). The explosive growth of social media has further amplified this threat, because the abundant multimedia content on such platforms is a rich source for extracting valuable voice samples. To illustrate the severity of audio-deepfake attacks, consider that a recent audio-deepfake-based cybercrime resulted in a 35 million USD loss for a UAE-based company.[2]

Given its research challenges and impact, the scientific community has been investigating methods to detect audio forgeries and defend against such attacks. For example, previous studies developed machine learning (ML) and deep learning (DL) models that can distinguish between fake and authentic speech (Abdel-Hamid et al., 2014; Camacho et al., 2021; Huang & Pun, 2020; Lv, Zhang, Tang, & Hu, 2022; Wang et al., 2020). These approaches have achieved promising results in terms of detection accuracy; however, they remain vulnerable to adversarial attacks. Adversarial attacks are designed to fool DL detectors by automatically producing a synthetic fake audio sample thatcan be classified as real using a legitimate classifier. Thus, given the damage potential of deepfake audio attacks, the underlying detectors must be resilient to such adversarial attacks.

* Corresponding author.
E-mail addresses: mora33056@hbku.edu.qa (M. Rabhi), spiridon.bakiras@singaporetech.edu.sg (S. Bakiras), roberto.dipietro@kaust.edu.sa (R. Di Pietro).

[1] The code of the attacks will be released open-source in the camera ready.

[2] https://tinyurl.com/4m88776u

https://doi.org/10.1016/j.eswa.2024.123941
Received 8 January 2024; Received in revised form 3 April 2024; Accepted 6 April 2024
Available online 9 April 2024

Previous studies have demonstrated the general vulnerability of neural networks to adversarial attacks (Carlini & Wagner, 2017). In contrast, in the current study, we seek to explicitly highlight this concern in the context of audio-deepfake detection. Several AI-based solutions have been proposed to detect deepfake audio (Lv et al., 2022; Wang et al., 2020); however, few studies have focused on investigating the specific vulnerabilities of these solutions. In particular, we assess the vulnerability of state-of-the-art deepfake detectors and demonstrate that adversarial attacks can be used to bypass fake audio-detection systems. Specifically, we perform an adversarial attack against a state-of-the-art classifier, Deep4SNet (Ballesteros et al., 2021) which is a deepfake audio detector that leverages a two-dimensional (2D) representation of the audio and convolutional neural networks (CNN) to classify fake audio. By leveraging the classifier's vulnerabilities, we discovered that even nonmeaningful adversarial examples can easily fool the detector and be classified as legitimate. Note that the code for the proposed attacks will be published in the camera-ready version of this paper. We also propose a general, low-cost, and straightforward solution for audio-deepfake adversarial attacks. The proposed defense mechanism is general, ensuring compatibility across a variety of deepfake audio detection models. Moreover, the proposed defense mechanism is characterized by its low-cost implementation, achieving two results: i. it alleviates the tool on resources typically linked to the deployment of complex defense mechanisms; ii. it might be used in contexts characterized by a generally lack of resources for implementing security mechanisms, hence enlarging the perimeter of access control. Finally, the straightforward nature of our solution facilitates its integration into existing audio deepfake detection frameworks, minimizing the need for extensive reconfiguration or specialized expertise.

The primary contributions of the study are summarized as follows:

- We demonstrate the vulnerability of DL-based deepfake audio detectors to adversarial attacks.
- We introduce two successful adversarial attacks on an audio-deepfake classifier using generative adversarial networks (GANs). The first attack attempts to generate adversarial examples starting from random noise and the second attack attempts to transform a fake speech sample to bypass the detector.
- After analyzing the audio deepfake detectors' vulnerabilities, we propose a generalized, low-cost, straightforward, and lightweight solution to defend against such adversarial attacks. The proposed solution is applicable to any fake audio classifier.

The remainder of this paper is organized as follows. Section 2 summarizes the existing literature on audio deepfake detectors, and Section 3 presents adversarial attacks on the state-of-the-art audio-deepfake detectors. Section 4 introduces the experimental methodology and discusses the results of the evaluation. A defense mechanism to mitigate adversarial attacks on audio deepfake detectors is proposed in Section 5 and the proposed defense method and results are discussed in Section 6. Finally, the study is concluded in Section 7.

2. Related work

In the following, we summarize existing fake audio-detection models and describe the different types of attacks on the classifiers.

2.1. Fake audio-detection methods

Deepfake audio is generated, edited, or synthesized, which is typically obtained via AI techniques and is difficult to distinguish from real audio. Detecting deepfake audio is crucial because it can have harmful effects in various fields. Existing studies on fake audio detection can be categorized into ML and DL models. In the ML domain, Rodriguez-Ortega et al. (2020) proposed a model based on logistic regression for detecting fake audios. Subsequently, they created a fake audio dataset using the imitation method by extracting the entropy features of real and fake audio. Then they trained their ML model using this dataset and achieved an accuracy of 98%. Although the model achieves high accuracy, the data must be preprocessed manually to extract the relevant features used by the model. Similarly, Singh et al. (2021) introduced a quadratic SVM model to distinguish between synthetic audio and authentic human voice. The model was trained on a dataset comprising both AI-generated audio (fake audio) and human speech data. The model achieved a classification accuracy of 97.56%, with a misclassification rate of 2.43%.

Li et al. (2021) introduced Res2Net, which is a modified ResNet architecture to detect fake audio. Res2-Net leverages different acoustic features to evaluate the model; however, the authors claimed that the best performance is obtained with CQT features. The proposed model demonstrated good performance in detecting fake audio; however, its generalizability requires further improvement. Gao et al. (2021) proposed a synthetic speech detection method based on the long-range frequency analysis of log-mel spectrograms. Specifically, they use the global 2D-DCT features to train a residual network that detects manipulated speech. The resulting model demonstrated good generalizability; however, its performance degraded when utilizing noisy samples.

Note that the aforementioned models rely on manually extracted features. More importantly, they require intensive preprocessing prior to training the ML models, which is a time-consuming process that can lead to inconsistencies. Therefore, previous studies have developed DL models for fake audio classification. For example, Camacho et al. (2021) transformed the input audio into scatter plot images of neighboring samples. The transformed input is then input to a CNN model trained on the Fake-or-Real (FoR) dataset. They reported an accuracy of 88.9% and demonstrated that their model has good generalizability when trained on different datasets. However, the model's accuracy is rather low compared to existing models, *e.g.*, the equal error rate was 11%.

Gomez-Alanis et al. (2019) proposed a light convolutional gated recurrent neural network to extract features from the audio signals and classify real and fake audio speech data. Here, an RNN was employed to extract long-term features, and a CNN was used to extract discriminative features. The model is trained on the ASVspoof 2019 dataset and exhibited good accuracy. However, this model is not generalizable to real-world examples. In addition, Subramani et al. (2020) proposed two CNN models for detecting fake audio: EfficientCNN and RES-EfficientCNN. Here, the input audio was first transformed into a spectrogram prior to being input to the CNN models. These models were tested on ASVspoof 2019 data and demonstrated good detection results. They demonstrated that RES-EfficientCNN model performed slightly better, with an F1-score of 97.61 compared to EfficientCNN's 94.14.

Lataifeh, Elnagar, Shahin, and Nassif (2020) proposed a model to discriminate between authentic Quran recitation and imitations. This method deploys both CNN and BiLSTM classifiers to detect imitations. Here, the audio files were first preprocessed and transformed into spectrograms before being input to the CNN and BiLSTM models. The CNN classifier achieved a higher accuracy rate than BiLSTM, with 94.33%. Aravind et al. (2020) proposed a fake audio classifier using transfer learning and the ResNet model. This model leverages a 2D representation of the audio to identify fake speech. First, the audio files are transformed into mel-spectrograms. Mel-spectrograms are a 2-D feature map in which one dimension represents time, the other represents frequency, and the values represent amplitude. The 2D features are then inputted to ResNet to classify the audio. The authors report an EER of 5.32% on the ASVspoof dataset; however, the training phase of this method incurs high time-consuming due to the deep network architecture.

Based on the models discussed above, we can conclude that although DL methods avoid manual feature extraction and excessive training processes, they still require special transformations for audio data. The discussed models are summarized and compared in Table 1.

Table 1
Summary of state-of-the-art models.

Model	Approach	Features used	Dataset	Challenges
Rodriguez-Ortega et al. (2020)	Logistic Regression	Time domain waveform	H-Voices	Manual Feature Extraction
Singh et al. (2021)	Quadratic SVM	MFCC, $\Delta - Cepstral$, $\Delta^2 - Cepstral$	own dataset	Manual Feature Extraction
Li et al. (2021)	Res2-Net	spectrogram, LFCC, CQT	ASVspoof	Generalization Improvement Needed
Gao et al. (2021)	Residual Network	Global 2D-DCT Features	ASVspoof 2019, Fake or Real (FoR), RTVCspoof	Performance Degradation with Noise
Camacho et al. (2021)	CNN	Extracted features from CNN on Scatter Plot Images	Fake or Real (FoR) Dataset	Lower Accuracy Compared to Some Models
Gomez-Alanis et al. (2019)	Convolutional Gated RNN	Spectogram	ASVspoof 2019	Lack of Generalization to Real-world Examples
Subramani et al. (2020)	EfficientCNN, RES-EfficientCNN	Spectrogram	ASVspoof 2019	Preprocessing to extract feature before classification
Lataifeh et al. (2020)	CNN and BiLSTM	Spectrogram	Arabic Diversified Audio	Lack of Generalization
Aravind et al. (2020)	Transfer Learning (ResNet)	Mel-Spectrograms	ASVspoof Dataset	Time-consuming Training
Ballesteros et al. (2021)	Deep4SNET	Histogram	H-voices	Data transformation preprocessing

2.2. Attack types

Adversarial attacks on neural networks fall into three categories, depending on the information available to the attacker. These three attacks are summarized as follows.

- *Blackbox attacks*: In this scenario, the adversary has no prior knowledge of the victim's model, *i.e.*, the attacker has no knowledge of the model architecture or parameters.
- *Whitebox attacks*: In this scenario, the adversary has complete access to the victim's model. In other words, the adversary possesses information about both the model architecture and the corresponding parameters.
- *Graybox attacks:* The distinction between white-box and gray-box attacks can be nuanced, depending on whether the scenario is examined from the attacker's or defender's perspective. From the defender's perspective, a gray-box attack occurs when the attackers have limited knowledge of the victim's model. The attacker can access its architecture, weights, and parameters; however, it is unaware of the model's defense mechanism or any added security layers. In contrast, from the attacker's perspective, if the attacker has complete insight into the classification model, including its architecture, parameters, and weights, the attack is frequently classified as a white-box attack. Several optimization-based attacks have been proposed, in which the attacker has full knowledge of the classifier and its parameters. For example, using our definition of graybox attacks, FGSM (Goodfellow, Shlens, & Szegedy, 2014), C&W, (Carlini & Wagner, 2017) can be used as a graybox attack since they assume complete knowledge of the target model's architecture, parameters, and weights. Previous studies have adopted adversarial attacks in the visual domain. For example, Iter, Huang, and Jermann (2017) employed FGSM (Goodfellow et al., 2014) to create adversarial MFCC features. Gong and Poellabauer (2017) proposed a gradient sign method that produces end-to-end audio adversarial examples based on an FGSM (Goodfellow et al., 2014). In the aforementioned attacks, the attacker starts from the original input and adds noise in order to create a synthetic input that fools the classifier. In contrast, the goal of the current study is to construct a synthetic input beginning from noise.

In this study, we focused on graybox attacks where the attacker has some knowledge about the victim model. Not that whitebox and gray-box attacks are obviously more powerful than blackbox attacks (Croce & Hein, 2020); thus, we opt for this type of powerful adversary with more destructive capacities to test the security limits of the victim model.

3. Adversarial attack

The primary objective is to demonstrate the vulnerability of fake audio detectors that leverage histograms to perform deepfake audio classification. Importantly, subsequent to the demonstration of the vulnerability of the state-of-the-art system, we propose a lightweight add-on defense mechanism that can enhance the reliability of the state-of-the-art model. However, our proposed defense mechanism is not limited to Deep4SNet; rather, it can be applied to any deepfake-audio detection system. Further details of the proposed mechanism are described in Section 5. In this section, we engineer an attack on a state-of-the-art audio deepfake detector, namely Deep4SNet (Ballesteros et al., 2021). This engineered attack employs GANs to build (synthetic) samples expected by the classifier. Such samples can fool the deepfake audio-detection system and be classified as authentic samples. In the following discussion, we refer to a system under attack as the victim model.

3.1. Victim model

Deep4SNet is an audio-deepfake detection model developed by Ballesteros et al. (2021). The core concept behind Deep4SNet is to transform a fake audio-detection task into a computer-vision task by transforming the input audio data into corresponding histogram images. The rationale behind this approach is that CNN-based classifiers are known to be highly accurate in image classification tasks. Thus, the authors proposed a CNN model classifying the deepfake audio. The model takes audio histogram images as input and determines whether the corresponding audio is fake or real. The Deep4SNet classifier was trained on a dataset of histograms representing real audio recordings and fake audio recordings obtained using imitation and Deep Voice. Deep4SNet achieved an accuracy of 98.5% when classifying fake speech. The model also achieves high precision (P) and recall (R) scores. In particular, $P = 0.997$ and $R = 0.997$ when using imitation-based fake audios, and $P = 0.985$ and $R = 0.944$ were obtained using Deep Voice fake audios.

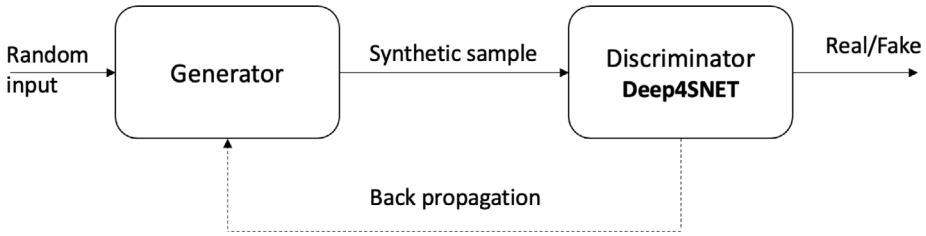


Fig. 1. Illustration of adversarial attack model architecture in deepfake-audio detectors.

3.2. Threat model

In this section, we describe the threat model used in this work to assess the effectiveness of the state-of-the-art model, Deep4SNet. In this study, we assume a graybox attack, where the adversary is aware of the architecture and parameters of the classifier; however, the attacker has limited knowledge about the defense mechanism (if any). Intentionally limiting the attacker's knowledge makes the threat more complicated, resembling the uncertainties that the adversary encounters in the real world scenarios. Note that the attacker also has access to the training data used by the classifier.

In adversarial attacks, attackers generally have two objectives: either they seek to reduce the overall accuracy of the target model (referred to as untargeted attacks), or they focus on compromising the classification of a specific class (referred to as targeted attacks. In this study, we assumed that the attacker's goal is to perform untargeted attacks.

3.3. Attack

Given the victim model, the target task is to generate synthetic histograms that can fool the deep neural network and be classified as legitimate audio. Our proposed attack model leverages GANs to generate such histograms.

GANs were first introduced by Goodfellow et al. (2020) and compromise a Generator G and a discriminator D. The generator and discriminator compete against each other, where the generator G generates examples, while the discriminator D attempts to classify whether the generated examples are real or generated. GANs have achieved good results in image generation and manipulation.

The overall architecture of the proposed GAN attack is shown in Fig. 1. The generator constructs histograms that resemble legitimate histograms constructed from real audi data and the discriminator classifies the histograms into Real or Fake. In this attack, we employ the trained victim model, Deep4SNet, as a discriminator, and we train the generator to build synthetic histograms that can fool Deep4SNet.

In this study, the following two attack scenarios were considered:

- **First attack scenario:** In this scenario, we trained the attacker to generate histograms starting from random noise. The objective is to produce histograms from a random noise input that appears legitimate, fool the discriminator, and be classified as Real. This attack scenario involves simulating a situation in which the adversary generates seemingly genuine audio by starting with completely random noise. This exposes a vulnerability in the Deep4SNet classifier in differentiating between authentic and manipulated content.
- **Second attack scenario:** In this attack scenario, the attacker begins with a specific audio sample, *e.g.*, a voice command. Herein, we attempted to explore the vulnerability of Deep4SNet to GAN-generated adversarial noise, specifically tailored from a known audio sample. To this end, we first created a voice command by cutting and pasting words from different audio speeches of the same person. For instance, an audio command that can be used as input is "Open the safe, please". Next, we transformed the resulting audio command into a histogram that we feed into the generator. The goal is to map the fake input histogram to a histogram that misleads the discriminator such that it cannot distinguished from real images in the original class and mistakenly classifies it as authentic.

3.3.1. Discriminator architecture

The discriminator used in this case is the victim model, *i.e.*, Deep4SNet, whose architecture is shown in Fig. 2. The CNN model comprises three convolutional and max-pooling layers, followed by a flattened layer, a hidden layer, and an output layer. The goal of the discriminator is to classify the original and fake speech recordings.

3.3.2. Generator architecture

The generator G_{attack} is constructed to learn the distribution of the target model. The problem can be formulated as follows:

$$f_{target}(G_{attack}(z, y)) = y_t \tag{1}$$

where f_{target} represents the target model to be attacked, that is Deep4SNet, z is the input of the generator, y is the true label of the generated image, and y_t is the target class of the classifier f_{target}, *i.e.*, Real. The generator G_{attack} is designed to generate histograms that can fool Deep4SNet model. To achieve this goal, we construct a generator using a stack of deconvolutional layers. The deconvolutional layers transfer a few features to complex examples, and represent the inverse operation to convolution. Deconvolutional layers have been efficient in practical image generation.

In the first scenario, the attacker generates histograms starting from random noise. The generator takes random points from the latent space as input, where the latent space is an arbitrarily defined vector space of Gaussian distributed values. These points represent a random input z that is passed to the generator $G_{attack1}$ in order to build histograms. The generated histogram's shape will be the same as the victim's model input histogram, *i.e.*, $150 \times 150 \times 3$. The generator's architecture used in the first attack scenario is illustrated in Fig. 3.

First, the random vector z is passed to a dense layer that transforms the vector to a lower representation of the output image, $15 \times 15 \times 3$. Next, we upsample the lower-resolution images to a higher resolution. For this reason, we use three Conv2DTranspose layers with 64, 32, and 32 filters, respectively, and a kernel size of 5×5. The LeakyRelu activation function is used for the first three Conv2DTranspose layers. Finally, the output layer has three filters for the three required channels.

The generator $G_{attack1}$ is not trained directly; instead, it is trained via the discriminator model. In this work, we adopt the gray box scenario attack, *i.e.*, we assume that the attacker has complete access to the detector model, including the architecture and parameters of the classification model. However, the attacker is not aware of the defense mechanism, if any. To construct our adversarial attack using the model described above, the generator aims to minimize the following loss function:

$$loss_G(z) = log(1 - D(G_{attack1}(z))) \tag{2}$$

where D represents the discriminator Deep4SNet.

Attack scenario two was more targeted. In other words, the attacker begins with a specific phrase—in our case: "Open the safe, please".

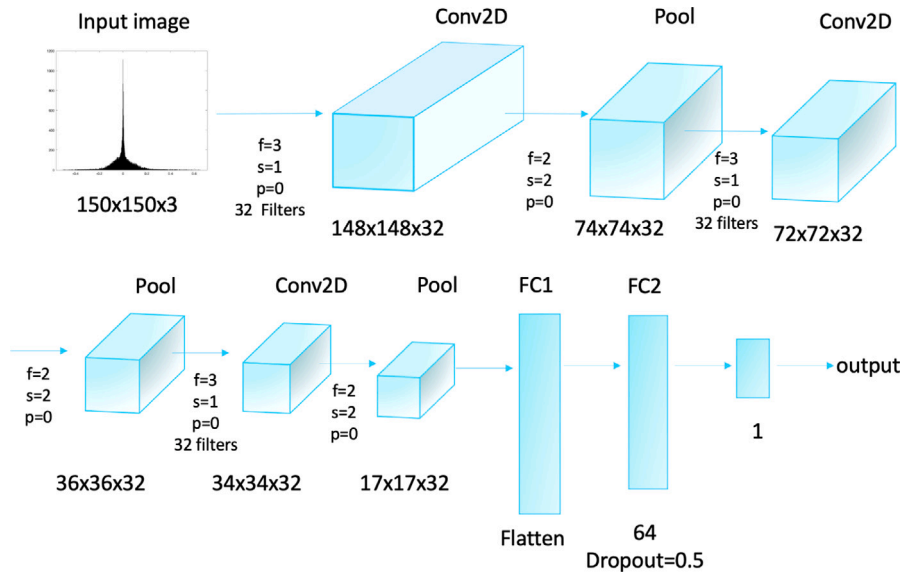


Fig. 2. Victim's model architecture.

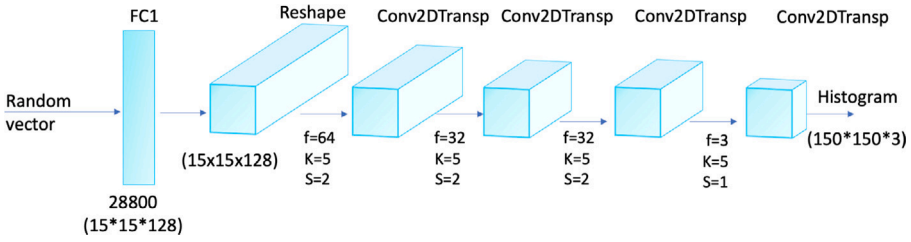


Fig. 3. Architecture of generator in attack scenario 1.

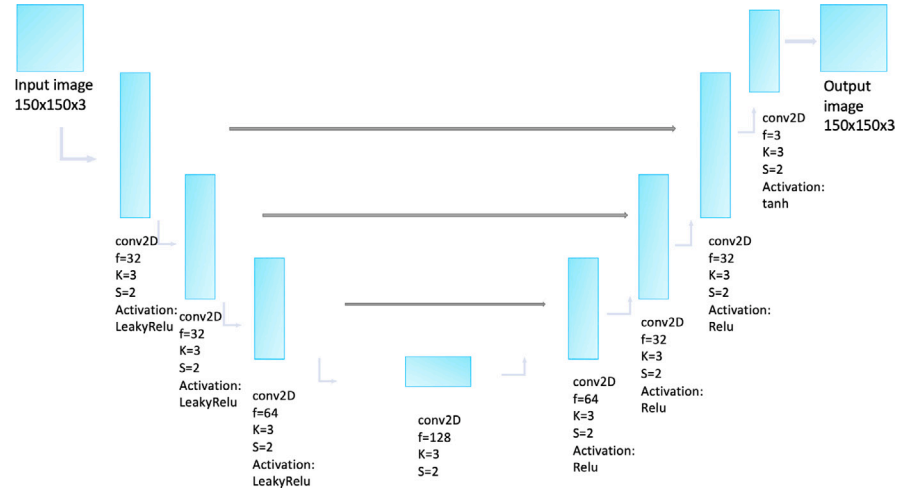


Fig. 4. Generator architecture under the second attack scenario. The gray arrows represent the concatenation between the corresponding encoder and decoder blocks.

Phrase audio is obtained by cutting and pasting words from different speeches of the victim. We then create a histogram, starting from the obtained fake audio. The resulting histogram is used as an input to the attacker's generator, named $G_{attack2}$. The goal of the attacker is to map the fake input histogram into a histogram that bypasses the classifier by building a generator that transforms the fake histogram into a histogram that can fool the detection model. To this end, the neural network selection was made experimentally.

We found that the generator architecture in Pix2Pix GAN proposed by Isola et al. (2017) obtained the best performance under the second attack scenario. This generator was based on a modified UNet architecture (Ronneberger et al., 2015), as illustrated in Fig. 4. It takes a source image as input and generates a target image that is able to fool the discriminator. To do this, it encodes the input image down to a bottleneck layer, and then it decodes the bottleneck representation to the size of the output image. The encoder is constructed of Conv2D layers followed by BatchNormalization layers. The decoder comprises a Conv2DTranspose layer followed by a dropout of 30%. Note that the final block of the encoder does not have a dropout layer. As per the UNet architecture, the output of the encoder blocks is concatenated

after the corresponding Conv2DTranspose layers in the decoder blocks, which helps the generator reconstruct the original image.

The generator $G_{attack2}$ is trained using the discriminator Deep4SNet. At every step, $G_{attack2}$ was updated to minimize the loss predicted by the discriminator for the generated histograms marked as legitimate. As such, it is encouraged to generate more realistic histograms. The generator $G_{attack2}$ employs the cross-entropy loss function described in Eq. (2).

4. Experimental results

In this section, we discuss the implementation of the described GAN attacks and discuss the obtained results. The results demonstrate that the proposed attacker models can fool the state-of-the-art solution for audio-based authentication tasks.

4.1. Dataset

In this study, we focus on graybox attack; thus, we assume that the attacker can access the data used for training the detector. Therefore, to train the generator, we employed the same dataset used by Ballesteros et al. (2020) to train Deep4Snet. This dataset contained 6,672 histograms of original and fake audios. Since our interest is to generate histograms that can be labeled as Real by Deep4SNet, we train our generators using only the set of histograms representing legitimate audios from the training set, *i.e.*, a total of 2,020 histogram images. The images are resized to $150 \times 150 \times 3$ and normalized to the range $[0, 1]$.

4.2. Experimental setup

Tensorflow and Keras 2.10.0 were leveraged as the deep learning frameworks to implement the GAN. The proposed models were trained using a desktop equipped with 64 GB RAM, 16 core processor AMD Ryzen 9, and Nvidia GeForce RTX 2080 Ti GPU running Windows 11 Pro, version 21H2. When run on the GPU, the attacks took approximately 1.38 s and 14 s per epoch for the first and second attack scenario, respectively. The Adam optimizer was used with the learning rate set to 10^{-4} and $2{\times}10^{-4}$ for the first and second experiments, respectively. In addition, the exponential decay rate for the first estimate is set to 0.9 and 0.5 for the first and second attack scenario, respectively.

4.3. Experimental results

In this section, we present the results of the Deep4SNet model attack under the two aforementioned attack scenarios.

4.3.1. Attack scenario 1

First, we evaluated the quality of histograms generated from random noise. We trained the generator for 2,000 epochs. Fig. 5 shows several samples of the generated histograms that successfully fooled the audio-deepfake detector, *i.e.*, the images were classified as authentic audio by Deep4SNet. Some images do not represent a valid histogram image as shown in Fig. 5(a) and Fig. 5(b). Furthermore, some generated histograms are not continuous, *e.g.*, one frequency in the histogram has more than a count, as in Fig. 5(d), which cannot be the case in a legitimate histogram. Therefore, despite the aforementioned histograms being unable to represent authentic speech, they were classified as legitimate audios by Deep4SNet. These findings suggest that the performance of Deep4SNet is significantly reduced in the presence of an adversarial attack, even when the attacker uses a simple generator.

In addition, to better assess the performance of Deep4SNet under adversarial attacks, we used the trained generator, $G_{attack1}$, to generate 100 histograms (starting from random noise), and evaluated the attack success rate. The attack success rate provides a quantifiable measure of the extent to which adversarial examples generated by $G_{attack1}$ successfully evade the target model Deep4SNET. This experiment was repeated 100 times, and the results demonstrate an average success rate of 99.91%, which further confirmed that the detection accuracy of Deep4SNet decreased drastically under adversarial attacks and reached near 0% accuracy (from 98.5% to 0.08%). This reduction in accuracy highlights the vulnerability of Deep4SNet to adversarial manipulations, emphasizing the need for robust defense mechanisms to enhance its resilience against such attacks.

Deep4SNet's vulnerability to adversarial attacks stems from its reliance on audio histograms. In fact, a generated Real histogram can be mapped to any number of noise or real audio samples, implying that Deep4SNet can classify random noise audio as authentic. In addition, a speech sample and its reversed version will have the same histogram representation, and both be classified as legitimate audio, even though reversed speech is neither meaningful nor authentic. Therefore, based on our previous findings, relying only on histogram representations for detecting fake audio lacks robustness.

4.3.2. Attack scenario 2

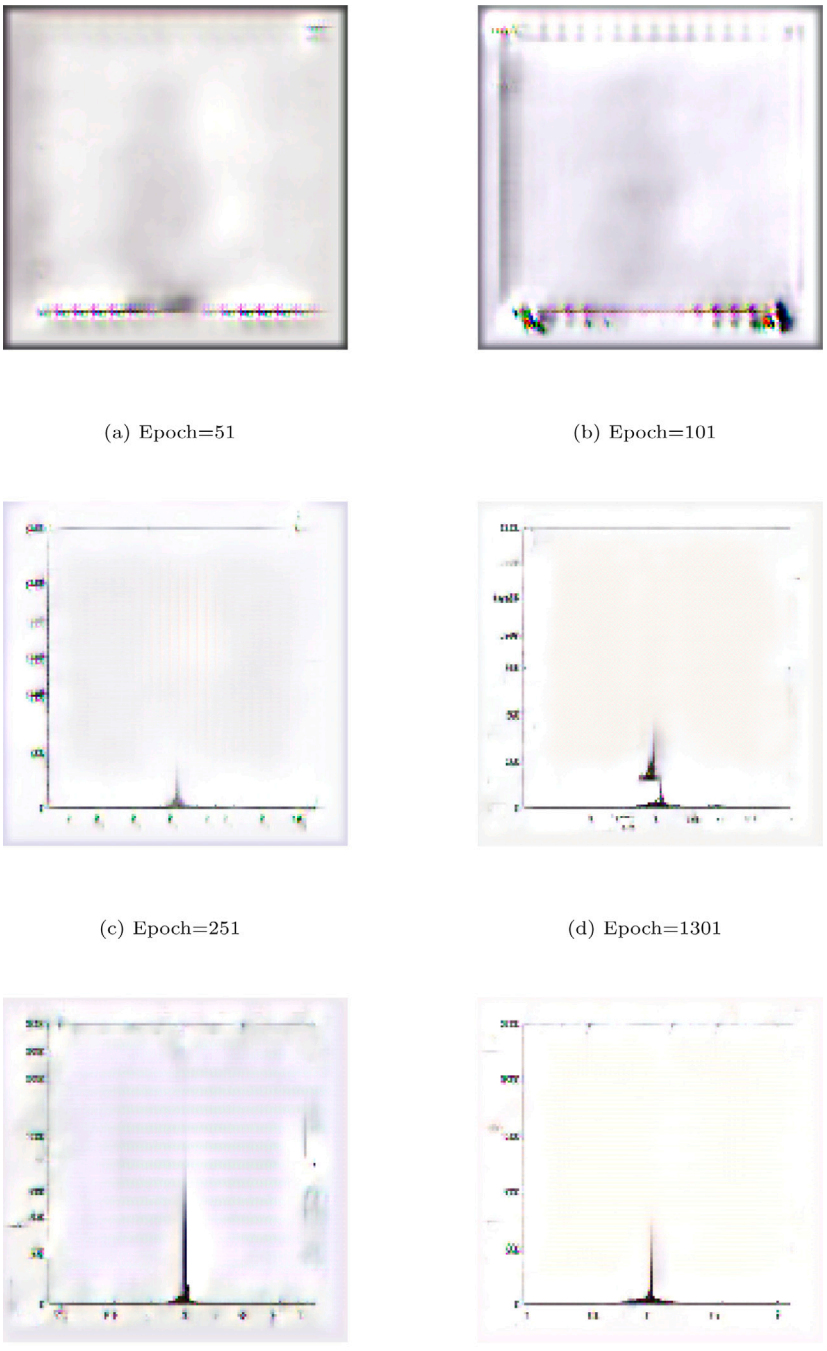
We constructed the second adversarial attack starting from a histogram representing fake audio. Similar to the first attack, we trained the generator $G_{attack2}$ for 2,000 epochs and successfully generated several histograms that can fool the Deep4SNet detector. Fig. 6 shows examples of the histograms obtained using $G_{attack2}$. As in the first attack, we observe that some discontinuous histograms, *e.g.*, those shown in Fig. 6(c), 6(e) and 6(f), were able to fool the Deep4SNet detector and be classified as legitimate audio. These discontinuous histograms do not present an authentic histogram since one frequency has more than one count. Furthermore, note that there is not much training required to build an adversarial example that fools Deep4SNet. For instance, already at epoch 151, the generator produced a histogram that was detected as authentic.

To further evaluate the performance of Deep4SNet under attack scenario 2, we generated 100 histograms using the trained generator $G_{attack2}$, starting from a fake histogram, and repeated the experiment 100 times. The aim of this experiment is to map the input fake histogram to a valid histogram that can fool the detector. We input the resulting 100 images into the Deep4SNet detector and observed an average success rate of 96.3%, indicating that a significant majority of the generated histograms managed to deceive the Deep4SNET detector successfully. These results confirm that Deep4SNet is vulnerable to generative attacks. Moreover, these findings further confirm that Deep4SNet is vulnerable to adversarial examples; thus, we must question the use of Deep4SNet and underscore the critical need for enhancing the robustness of Deep4SNet.

We demonstrated that, under both adversarial attack scenarios, we were able to bypass Deep4SNet detectors easily without the need for high computing power or advanced tools. This is a significant security concern because audio deepfake detectors can be used to authenticate users. More importantly, we showed that using histograms as the underlying detection mechanism for audio deepfakes is not resilient to attacks. In fact, a histogram is not a unique representation of an audio frame and can be mapped to different audios that contain either speech or random noise, which is a feature leveraged in our GAN-mounted attack.

In this study, we selected Deep4SNet to demonstrate the vulnerability of DNN-based audio-deepfake detectors to adversarial attacks. However, such attacks are not exclusive to Deep4SNet and can be designed to attack any audio deepfake detector that employs DNNs. Specifically, one would only need to adapt the generator architecture to the new discriminator, *i.e.* a deepfake detector, to construct an effective adversarial attack against this new detector.

Finally, our work motivates the need to incorporate defense mechanisms against adversarial attacks when designing DL-based deepfake detectors or any DL-based classifier in general. In fact, several fake audio detectors in the literature do not consider adversarial attacks



(a) Epoch=51 (b) Epoch=101

(c) Epoch=251 (d) Epoch=1301

(e) Epoch=1651 (f) Epoch=1801

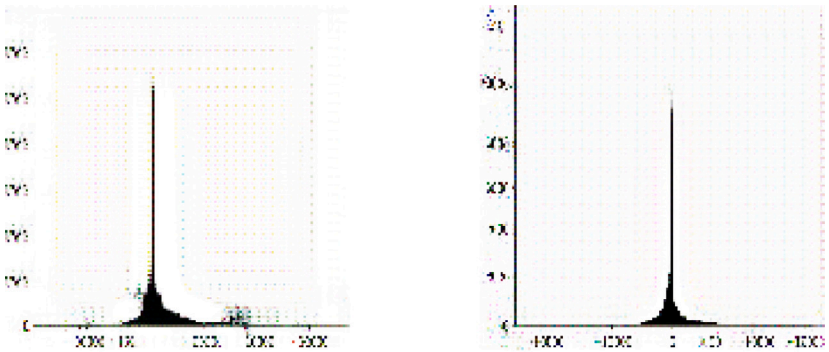
Fig. 5. Examples of generated histograms during generator training at different epochs under first attack scenario.

when designing their models or even discuss the security of the corresponding models. This questions the utility of such detectors because their detection accuracy can be reduced to 0% under attack. Therefore, we emphasize the importance of security analysis when designing DL-based detectors.

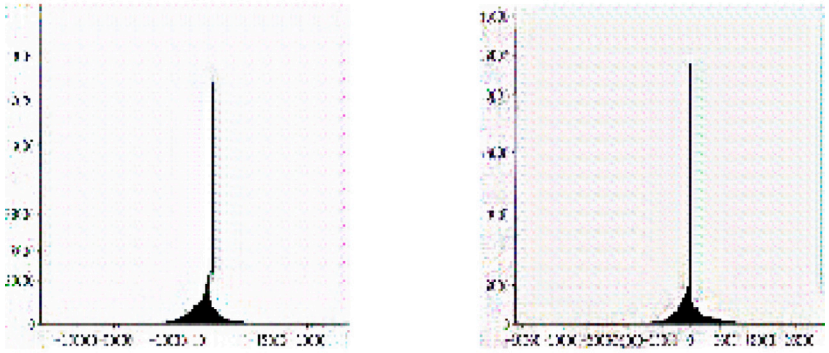
Table 2 summarizes existing solutions that neither analyze the security of their models nor test the models against adversarial attacks.

5. Proposed defense mechanism

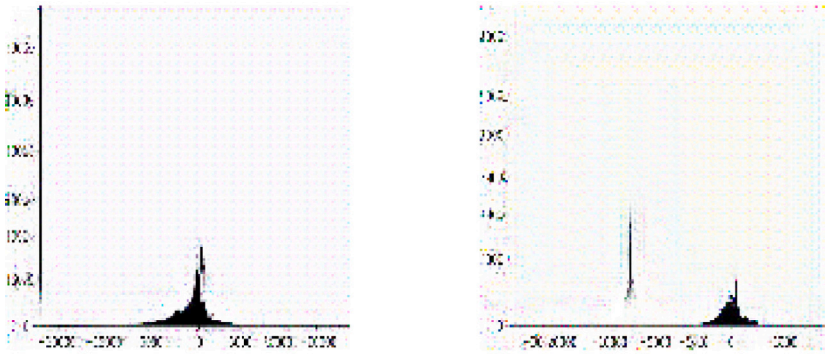
Having demonstrated the vulnerability of audio deepfake detection to adversarial attacks, we proposed a simple yet effective defense mechanism that can make the detector more resilient. The proposed solution offers a comprehensive, cost-effective, and highly efficient approach to addressing the challenges posed by generative attacks, ensuring broad



(a) Epoch=151 (b) Epoch=451



(c) Epoch=501 (d) Epoch=901



(e) Epoch=1601 (f) Epoch=1901

Fig. 6. Examples of generated histograms during generator training at different epochs under second attack scenario.

applicability and robust defense to deepfake audio detection models. This new detection framework is shown in Fig. 7. As can be seen, an independent speech-to-text layer is added to the initial detector to mitigate adversarial attacks. The core concept behind the framework is rather simple; if the audio is classified as Real by the deepfake detector, the speech-to-text layer verifies whether the audio frame matches the underlying text. If the audio does pass the Speech-to-Text add-on layer, then it will be considered Real audio. In case it fails to pass the speech-to-text add-on layer, the audio is determined to be fake. Thus, we can ensure that the limitations of the Deep4SNet detection model are overcome. Note that this concept can be applied to any detector and that the speech-to-text layer alone is not the proposed solution. Instead, the proposed framework involved two key steps, which are summarized as follows.

- **Step 1:** The first step involves preliminary verification of the legitimacy of the given audio using an audio-deepfake detector. The output is the initial classification of the input audio.
- **Step 2:** If the audio file is provisionally labeled as Real by the detector, it is then passed to the speech-to-text analysis layer to further verify its authenticity. For instance, we use the speech-to-text API from Google, which transcribes the audio into its corresponding text if the audio is meaningful; otherwise, it will

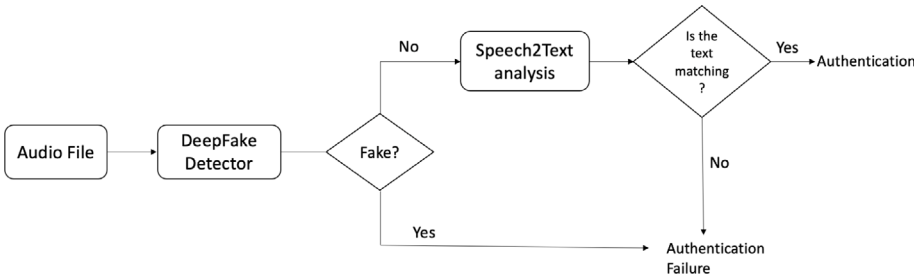


Fig. 7. Proposed framework to mitigate adversarial attacks on audio deepfake detectors.

Table 2
State-of-the-art deepfake detectors that do not present a security analysis of their solutions.

Ref	Technique	Features
Aravind et al. (2020)	ResNet	Spectrogram
Subramani et al. (2020)	CNN	Spectrogram
Camacho et al. (2021)	CNN	Scatter plots
Ballesteros et al. (2021)	CNN	Histograms

raise an error. If the API fails to produce a valid text output, the audio is classified as Fake. Thus, the Speech-to-Text layer checks whether the input audio corresponds to the expected text. When used alone, the Speech-to-Text would be a poor defense mechanism against fake audio attacks because any audio generating the expected text would pass the check.

In the following, we demonstrated how the proposed solution dramatically enhances the resilience of the Deep4SNet model. In the case where the adversarial attack starts from noise, the constructed input may fool the Deep4SNet detector in Step 1. However, in Step 2, the audio is translated to a nonmeaningful text, causing the system to reject the input audio and classify it as Fake. Note that attack scenario 2 is trickier than 1 because the attacker already started from a meaningful human-generated sentence. The adversarial examples might pass the first test successfully and be classified as legitimate. However, the attacker does not know the exact secret, *e.g.*, "Open sesame", to pass the detector. Even though the input audio is translated into a meaningful sentence, in Step 2, the sentence is compared to the original (expected) one. Hence, the attacker will not be able to pass the second step, and the adversarial example will be classified as Fake.

We conducted two experiments to prove the efficiency of the proposed defense mechanism. The goal of the first experiment was to prove that the proposed defense mechanism correctly classifies histograms that represent noise as fakes. To this end, we created audio noises, transformed them into histograms, and inputted them into the Deep4SNet model and Deep4SNet+Speech-to-text models. Figure 8 shows a few examples of histograms corresponding to white noise. These histograms, representing white noise, have successfully bypassed Deep4SNet and were classified as real audios. However, when using the proposed defense mechanism, while the audios still bypassed the first layer of defense (*i.e.*, Deep4SNet), when the corresponding audio was transcribed using the Speech-to-Text API, we noticed that the transcription output was empty since the audio did not contain any meaningful words. Hence, the audio was classified as Fake audio, preventing the attack from succeeding. The implementation of our proposed defense mechanism has augmented the effectiveness of Deep4SNet's classification. Indeed, despite the persistence of adversarially crafted audio samples in evading Deep4SNet, the additional layer of security introduced by our defense mechanism, *i.e.*, Speech-To-Text, has successfully detected the manipulated audio as fake.

Table 3
Attack success rate of white noises on victim model and proposed defense model.

	Deep4SNet	Speech-to-Text	Deep4SNet+Speech-to-Text
Attack Success Rate	91.1%	6.9%	0%

Table 3 shows the attack's success rate on Deep4SNet+Speech-to-Text and Deep4SNet when using noise samples. The attack success rate is used as a performance metric for our proposed defense mechanism to gauge its effectiveness in mitigating adversarial attacks. A reduction in attack success rate subsequent to the implementation of our defense strategy signifies its robustness in fortifying Deep4SNET against adversarial manipulation. In this experiment, we created 1000 audio samples of 4s each, starting from an audio recording of noise with a length of 60 min, available in.[3] We generated the corresponding histograms and passed them to Deep4SNet and Deep4SNet+Speech-to-Text, respectively. From Table 3, we can observe that Deep4SNet was generally unable to detect the noise, resulting in an attack success rate of 91.1%. Speech-To-Text detection has yielded a significant reduction in the success rate of adversarial attacks to reach 6.9%. However, when we used Deep4SNet+Speech-to-Text, the attack success rate dropped to 0%, *i.e.*, none of the input noise recordings were able to bypass the proposed defense mechanism. This was the expected outcome because, even though the audio samples were able to bypass the first step of the proposed mechanism, none of them were transcribed to the key phrase that the system was looking for, *i.e.*, all of the audio traces were transcribed to empty phrases, since they did not make any sense. This result demonstrated that the inexpensive, straightforward add-on solution is able to defend against noise-based and human-generated audio attacks, improving the overall accuracy of Deep4SNet.

In the second experiment, we start with a short audio speech of 7 s selected from the LJSpeech dataset (Ito & Johnson, 2017), which contains short audio samples from a single speaker reading passages from different books. As a first step, we transformed a random audio sample and its time-reversed version into histograms, as shown in Fig. 9. We notice that, as expected, the two histograms are indistinguishable—an audio trace played from the beginning to the end would generate the same histogram when playing the same audio from the end to the beginning. Hence, the Deep4SNet detector classifies both audios as legitimate, *i.e.*, the meaningless reverse audio is also classified as real audio. When using the proposed defense technique, the reverse audio will be detected as Fake since its transcription using Speech-to-Text would generate either a meaningless transcript or the transcript would not match the original phrase. It is worth noticing that the only possible exception would be palindrome sentences. However, such sentences are easy to test and could be declared as not acceptable at the enrollment phase. As such, reversed audio samples will not be able to fool the new proposed framework and would be classified as Fake.

[3] https://tinyurl.com/2hrzmzvf

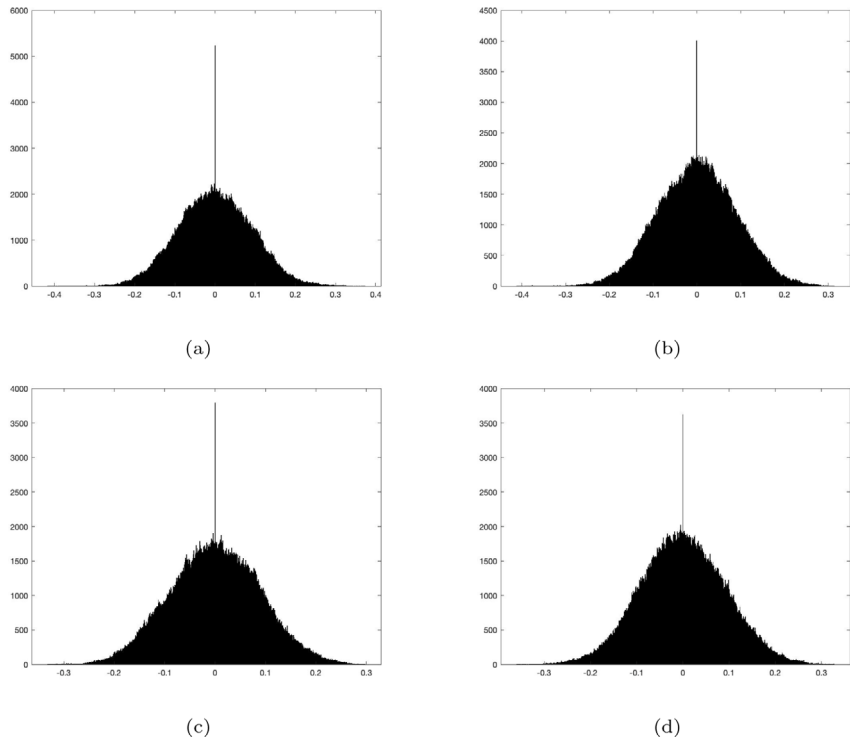


Fig. 8. Examples of histograms of noise audios of length 7s.

Table 4
Attack success rate of victim model and proposed defense model using authentic and reverse audio samples from LJSpeech dataset.

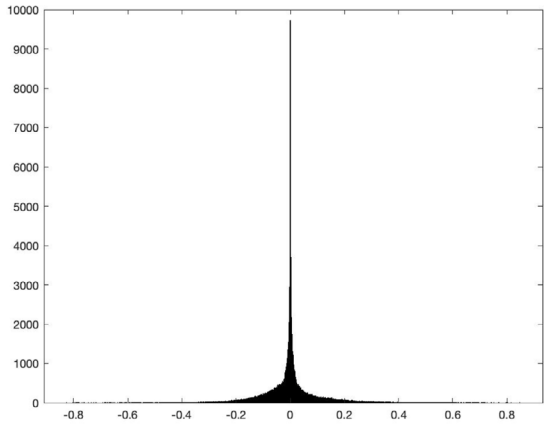
	Deep4SNet	Speech-To-Text	Deep4SNet+Speech-To-Text
Attack Success Rate	87.6%	53.4%	43.8%

To better evaluate the performance of the proposed method, we created a small dataset of 1000 samples, where 50% were selected randomly from the LJSpeech dataset, and the remaining 50% were created by reversing the previously selected 500 audio samples. The corresponding attack success rate results are shown in Table 4. Deep4SNet has classified 87.6% of the samples as Real audio samples, *i.e.*, the Deep4SNet model has authenticated 438 out of 500 legitimate audios as Real and equivalently 438 out of 500 reverse audio histograms as Real audio. In contrast, Speech-To-Text has classified 53.4% of the samples as authenticate audio samples. Remarkably, when employing the combined approach of Deep4SNet and Speech-to-Text, only 43.8% of the histograms were classified as real audio samples. This significant decrease in the attack success rate underscores the efficacy of the proposed defense mechanism in detecting adversarially manipulated audio samples, thereby enhancing the overall accuracy and reliability of the classification process. Moreover, we found that the accuracy of Deep4SNet has decreased when tested using another dataset (Ito & Johnson, 2017) , which suggests that the Deep4SNet model does not generalize well to other datasets. In addition, we notice that the attack success rate has reduced significantly using the proposed Deep4SNet+Speech-to-Text method to reach less than 50% (the rate of the legitimate audio samples in the generated dataset). The proposed defense technique was able to identify the reverse audios as fake: the reverse audios were able to bypass the first step—Deep4snet model—but the reverse samples did not manage to bypass the second step—Speech-to-Text layer. These results further confirm that the add-on Speech-to-Text layer helps reduce the impact of noise significantly, and limits the Deep4SNet classification to real speech samples.

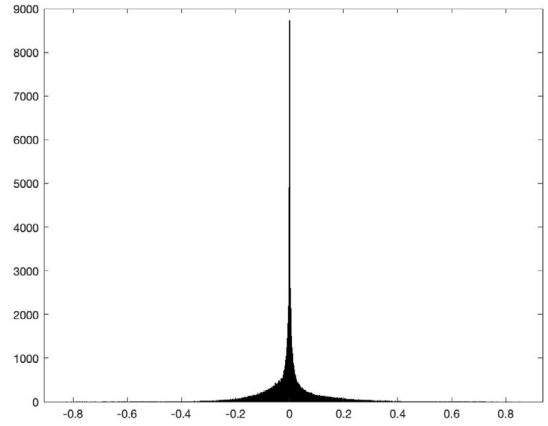
We have discussed the proposed defense mechanism in the context of the Deep4SNet classifier. However, this mechanism is not exclusive to the Deep4SNet detector. In other words, it can be generalized to any audio deepfake detector by adding the proposed Speech-to-Text analysis layer in series to an existing detector. This would make the classification decisions more accurate in the event of adversarial attacks and would reduce the impact of noise, such as the one generated by audio samples that do not have a syntactical structure.

However, previous studies have demonstrated that Speech-To-text systems can be vulnerable to adversarial attacks (Carlini & Wagner, 2018). In the context of the proposed system, we consider a graybox attack where the attacker is unaware of the defense system, *i.e.*, the Speech-To-Text. This obfuscation makes it difficult for the attacker to craft successful audio samples. Nevertheless, if we consider a white box attack, in which the attacker possesses comprehensive knowledge of the defense system, the attacker's challenge is accentuated as the crafted audio needs to deceive both the classification system and a Speech-to-Tex system, which adds a layer of complexity to the attack. In such a scenario, the attacker must consider the intricacies of both systems and devise an attack strategy that effectively bypasses the challenges posed by each. First, the attacker needs to create an audio sample that can deceive the classification model. Simultaneously, the attacker must also ensure that the crafted audio misleads the Speech-to-Text system. This involves manipulating the audio in a manner that does not compromise the ability to be transcribed into a desired target text, *i.e.*, the voice command. Hence, the attacker's challenge lies in striking a balance between the requirements of both systems. Changes made to deceive one system might inadvertently affect the other. Therefore, the proposed model introduces a multifaceted challenge for the attacker, thereby necessitating a sophisticated approach that considers the unique characteristics and vulnerabilities of each system in order to be able to fool both systems.

Finally, we would like to emphasize that we are aware that the reported examples provide straightforward cases where the proposed defense mechanism works efficiently and prevents the adversarial examples from being classified as Real audio samples. In fact, at this stage, we cannot formally exclude that an adversary could bypass



(a) Original audio



(b) Reversed audio

Fig. 9. A histogram example of a random audio file from the LJSpeech dataset and its reversed version.

such a countermeasure. Therefore, investigating a lower bound on the difficulty for an adversary to bypass such defenses is left for future work.

6. Discussion

To better understand the vulnerability of Deep4SNet, we further describe the problem as shown in Fig. 10. The problem of deepfake audio detection can be formulated as finding a set of authentic audio samples in the audio world. However, dealing with the audio files directly is not an easy task. Hence, many studies have suggested formulating the problem as a computer vision problem since deep neural network image classification has exhibited rapid advancements in the last decade particularly because of the ongoing development of CNN. In this study, we focused on using histograms to detect fake audio. In this context, we described the problem using three different sets: the audio world can represent any possible sound ($\mathcal{A}$), the histogram world ($\mathcal{H}$) is the set of any possible histogram, and the drawing world is the set of any possible drawings ($\mathcal{D}$). For instance, if we limit the audio samples to only 3 s length audio (in the 20 Hz–20 KHz, which is the frequency range that can be generally sensed by humans (Raponi, Oligeri, & Ali, 2022)), the audio world will include a very large but finite number of audio samples. These audio samples represent any possible human-hearable audio of length 3 s. Among these (very large but limited) possible sounds, only a subset would represent meaningful sentences. The audio-deepfake classifier is trained to learn the discrimination between this authentic subset and all the other sounds. Hence, it considers all the other possible audible sounds outside of the authentic speech set as Fake audio. Deep4SNet aims to learn this legitimate set as well by formulating the problem as a computer vision problem. It transforms the audio files into their corresponding histograms and tries to learn the discrimination between the histograms representing authentic audio and those representing fake audio.

However, when we attacked the Deep4SNet model, we found that the model could also identify a drawing in $\mathcal{D}\backslash\mathcal{H}$, *e.g.*, discontinuous histograms, images that do not show histograms, as legitimate histograms. This observation suggests that the victim model, *i.e.*, Deep4SNet, did not learn to classify drawings other than histograms as fake, and the training process only focused on histograms. Hence, it failed when it was tested using a set of drawings that do not represent a proper histogram, represented by the blue rectangle in the cited figure. That is, the set of elements that pass the authentication check ($\mathcal{P}$) comprises – at least – all the authentic audio, histograms that are generated by non-authentic audio, and by elements that belong to the drawing set, but not to the histogram set, as shown in Fig. 5(a), 5(b), 6(c), and 6(f).

Moreover, we have noticed that the classifier considers an audio file and its reverse as authentic histograms and hence these are classified as being authentic. While the audio file can be legitimate and represent an authentic speech, its reverse should not be detected as legitimate audio since it does not represent comprehensible audio—note that its transcription would not make sense. However, Deep4SNet is unable to detect it as such. This is due to the fact that histograms represent a count of the frequencies in the audio file without considering the

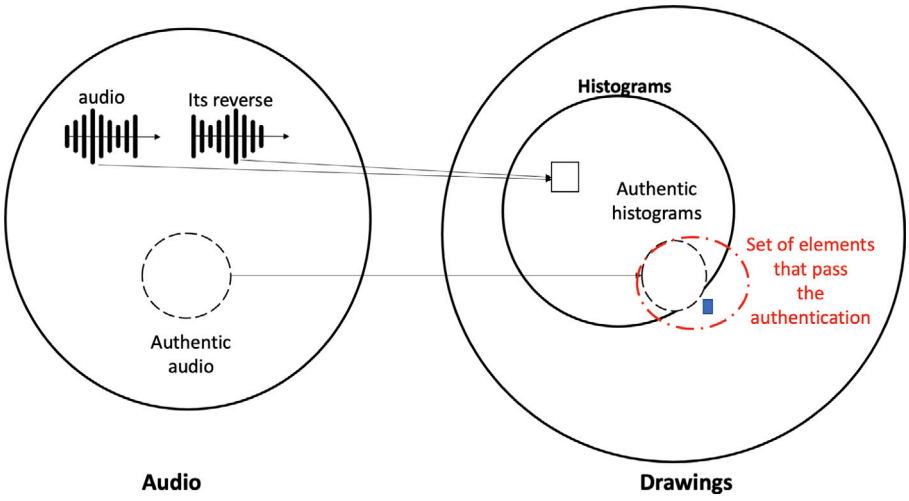


Fig. 10. A representation of domain and co-domain for the authentication challenge.

temporal order. Therefore, relying only on histograms to detect fake audio presents some critical limitations.

To overcome these limitations, we propose a defense solution that limits the mapping between the sound and the histogram worlds to only comprehensive legitimate speech. Using the proposed Speech-to-Text add-on, we can eliminate many fake samples that were previously considered legitimate by Deep4SNet, *e.g.*, reverse audio, drawings, etc. This will help the model achieve better discrimination between authenticated and fake audio and reduce the number of false positives.

Finally, we have shown that the proposed defense solution is beneficial to Deep4SNet and has significantly reduced the attack success rate. However, the solution works for other audio-deepfake classifiers as well. In future work, we aim to test the robustness of audio-deepfake classifiers that are using spectrograms and test the efficiency of the proposed solution using such classifiers.

7. Conclusion

GAN-based adversarial attacks are quite effective in DNN-trained models and can cause serious threats to DNN detectors. However, such attacks have not yet been addressed in the context of audio-deepfake detection. In this study, we attempted to fill this gap by demonstrating that a state-of-the-art audio-deepfake detector can be bypassed easily if, as commonly assumed in the literature, the adversary possesses knowledge of the detector's architecture and the dataset used for training. We designed two adversarial attacks based on the GAN architecture that generate fake histograms to fool the state-of-the-art Deep4sNet detector. The first attack, starting from random noise, was able to reduce the accuracy of the Deep4SNet detector from 98.5% to 0.08%, and the second attack, which aims to map fake audio to a histogram that fools Deep4SNet, reduced the accuracy of the detector to 3.7%. We demonstrated the effect of adversarial attacks on a specific state-of-the-art deepfake detector; however, our adversarial attacks can be generalized to other DNN-based detectors if the generator is adapted to the new detector. Finally, to mitigate these attacks, we propose a simple yet effective and inexpensive solution that implements a speech-to-text layer after the detector's classification decision to ensure that the input audio represents meaningful speech. As future work, we plan to expand the applicability of our proposed attacks beyond the current target model to encompass a wider array of deepfake audio detection systems. By extending the scope of our attacks, we aim to provide a more comprehensive understanding of their effectiveness across diverse audio deepfake detection models, thereby enhancing the generalizability of our attack. In addition, we aim to investigate the proposed defense technique further and evaluate its robustness when used in combination with other classifiers. Moreover, we plan to incorporate a trigger event solution into the proposed defense mechanism. Finally, we aim to implement an event-triggered strategy to monitor the histograms for anomalies or known adversarial patterns (Song, Wu, Song, & Stojanovic, 2023a; Song, Wu, Song, Zhang, & Stojanovic, 2023b).

The novelty of the context where the discussed GAN attack is reported, the generality of our findings and the provided discussion, coupled with the fact that the code of the attacks is released as open source, [4] other than assuring the reproducibility of results, will also foster further research in the field.

CRediT authorship contribution statement

Mouna Rabhi: Conceptualization, Methodology/Study design, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization, Project administration. **Spiridon Bakiras:** Conceptualization, Methodology/Study design, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration. **Roberto Di Pietro:** Conceptualization, Methodology/Study design, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Roberto Di Pietro reports financial support was provided by by the award Thematic Research Grant Program from Hamad Bin Khalifa University (HBKU). If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

[4] https://github.com/MounaRa/Audio-deepfake-detection-Adversarial-attack-and-countermeasures

Acknowledgments

This work was partially supported by the award Thematic Research Grant Program from Hamad Bin Khalifa University (HBKU), Office of the Vice President for Research, Doha, Qatar [VPR-TG01-009]. The information and views set out in this publication are those of the authors and do not necessarily reflect the official opinion of HBKU.

References

Abdel-Hamid (2014). Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *22*(10).

Aravind, P. (2020). Audio spoofing verification using deep convolutional neural networks by transfer learning. arXiv preprint arXiv:2008.03464.

Ballesteros (2020). A dataset of histograms of original and fake voice recordings (h-voice). *Data in Brief*, *29*.

Ballesteros (2021). Deep4SNet: deep learning for fake speech classification. *Expert Systems with Applications*, *184*.

Camacho, S. (2021). Fake speech recognition using deep learning. In *Workshop on engineering applications*. Springer.

Carlini, N., & Wagner, D. (2017). Towards evaluating the robustness of neural networks. In *2017 IEEE symposium on security and privacy*. IEEE.

Carlini, N., & Wagner, D. (2018). Audio adversarial examples: Targeted attacks on speech-to-text. In *2018 IEEE security and privacy workshops*. IEEE.

Croce, F., & Hein, M. (2020). Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*. PMLR.

Engel, J. (2019). Gansynth: Adversarial neural audio synthesis. arXiv preprint arXiv:1902.08710.

Gao, Y., Singh, R., & Raj, B. (2018). Voice impersonation using generative adversarial networks. In *2018 IEEE international conference on acoustics, speech and signal processing* (pp. 2506–2510). IEEE.

Gao, Y. (2021). Generalized spoofing detection inspired from audio generation artifacts. arXiv preprint arXiv:2104.04111.

Gomez-Alanis (2019). A light convolutional GRU-RNN deep feature extractor for ASV spoofing detection. In *Proc. interspeech, vol. 2019*.

Gong, Y., & Poellabauer, C. (2017). Crafting adversarial examples for speech paralinguistics applications. arXiv preprint arXiv:1711.03280.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, *63*(11).

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572.

Huang, L., & Pun, C. M. (2020). Audio replay spoof attack detection by joint segment-based linear filter bank feature extraction and attention-enhanced densenet-bilstm network. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *28*.

Isola (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Iter, D., Huang, J., & Jermann, M. (2017). Generating adversarial examples for speech recognition. Stanford Technical Report.

Ito, K., & Johnson, L. (2017). The lj speech dataset. 2017. URL https://keithito.com/LJ-Speech-Dataset.

Kumar, K. (2019). Melgan: Generative adversarial networks for conditional waveform synthesis. *Advances in Neural Information Processing Systems*, *32*.

Lataifeh, M., Elnagar, A., Shahin, I., & Nassif, A. B. (2020). Arabic audio clips: Identification and discrimination of authentic cantillations from imitations. *Neurocomputing*, *418*.

Li, X. (2021). Replay and synthetic speech detection with res2net architecture. In *ICASSP 2021-2021 IEEE international conference on acoustics, speech and signal processing*. IEEE.

Lv, Z., Zhang, S., Tang, K., & Hu, P. (2022). Fake audio detection based on unsupervised pretraining models. In *ICASSP 2022-2022 IEEE international conference on acoustics, speech and signal processing*. IEEE.

Oord, A., Li, Y., Babuschkin, I., Simonyan, K., Vinyals, O., Kavukcuoglu, K., Driessche, G., Lockhart, E., Cobo, L., & Stimberg, F. (2018). Parallel wavenet: Fast high-fidelity speech synthesis. In *International conference on machine learning*. PMLR.

Oord, A. v. d. (2016). Wavenet: A generative model for raw audio. arXiv preprint arXiv:1609.03499.

Raponi, S., Oligeri, G., & Ali, I. M. (2022). Sound of guns: digital forensics of gun audio samples meets artificial intelligence. *Multimedia Tools and Applications*, *81*(21).

Rodriguez-Ortega (2020). A machine learning model to detect fake voice. In *International conference on applied informatics*. Springer.

Ronneberger (2015). U-net: Convolutional networks for biomedical image segmentation. In *International conference on medical image computing and computer-assisted intervention*. Springer.

Singh (2021). Detection of AI-synthesized speech using cepstral & bispectral statistics. In *2021 IEEE 4th international conference on multimedia information processing and retrieval*.

Song, X., Wu, N., Song, S., & Stojanovic, V. (2023). Switching-like event-triggered state estimation for reaction–diffusion neural networks against DoS attacks. *Neural Processing Letters*, *55*(7), 8997–9018.

Song, X., Wu, N., Song, S., Zhang, Y., & Stojanovic, V. (2023). Bipartite synchronization for cooperative-competitive neural networks with reaction–diffusion terms via dual event-triggered mechanism. *Neurocomputing*, *550*, Article 126498.

Subramani (2020). Learning efficient representations for fake speech detection. In *Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 04*.

Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., & Bengio, S. (2017). Tacotron: Towards end-to-end speech synthesis. In *Interspeech 2017*. ISCA.

Wang, R. (2020). Deepsonar: Towards effective and robust detection of ai-synthesized fake voices. In *Proceedings of the 28th ACM international conference on multimedia*.