FraudWhistler: A Resilient, Robust and Plug-and-play Adversarial Example Detection Method for Speaker Recognition

Kun Wang
Zhejiang University

Xiangyu Xu
Southeast University

Li Lu[*]
Zhejiang University

Zhongjie Ba
Zhejiang University

Feng Lin
Zhejiang University

Kui Ren
Zhejiang University

Abstract

With the in-depth integration of deep learning, state-of-the-art speaker recognition systems have achieved breakthrough progress. However, the intrinsic vulnerability of deep learning to Adversarial Example (AE) attacks has brought new severe threats to real-world speaker recognition systems. In this paper, we propose *FraudWhistler*, a practical AE detection system, which is resilient to various AE attacks, robust in complex physical environments, and plug-and-play for deployed systems. Its basic idea is to make use of an intrinsic characteristic of AE, i.e., the instability of model prediction for AE, which is totally different from benign samples. *FraudWhistler* generates several audio variants for the original audio sample with some distortion techniques, obtains multiple outputs of the speaker recognition system for these audio variants, and based on that *FraudWhistler* extracts some statistics representing the instability of the original audio sample and further trains a one-class SVM classifier to detect adversarial example. Extensive experimental results show that *FraudWhistler* achieves 98.7% accuracy on AE detection outperforming SOTA works by 13%, and 84% accuracy in the worst case against an adaptive adversary.

1 Introduction

Speaker Recognition (SR) has been applied ranging from personalized services, digital forensics, to financial payments, in real-world scenarios. Various mature products thus come forth including voice assistants (e.g., Apple Siri, Microsoft Cortana, etc.) and voiceprint lock (e.g., HSBC's voice id and Barclays' voice security). A recent report [51] shows that the global voice biometric market size is expected to grow from $1319.23 million in 2021 to $4823.85 million by 2028. This is benefited from the advent of deep learning, and its in-depth integration into state-of-the-art SR systems achieving satisfactory performance. However, the intrinsic vulnerability of deep learning to Adversarial Example (AE) attacks [3, 6, 17] has brought new severe threats. Recent studies [2, 7, 9, 11, 53, 72] demonstrated that deep neural network-based speaker recognition could be spoofed by imposing subtle perturbations on benign utterances, which raises practical threats to real-world SR systems. To enable reliable SR systems, it is necessary to provide SR with a powerful defense against AE attacks.

Early studies [46, 59] investigate to make SR systems more robust by retraining the deep neural network model, which induces laborious efforts for deployed systems. To overcome it, the following work [8, 25, 62] proposed several pre-processing methods as filtering modules to purify audio samples. But such methods could be easily bypassed by adaptive attacks [56]. Another line of works [34, 61] focused on developing an effective binary detector trained with specific AE-generating algorithms. Unfortunately, such methods suffer from performance degradation on unseen AE attacks. Moreover, all of these methods are oriented to over-the-line defenses, without validating their effectiveness in over-the-air situations, to support wider scenarios, such as interactions with smart speakers.

Toward this end, our work aims to propose a practical AE detection method for SR, which is resilient on various AE generating algorithms, robust in complex physical environments, and plug-and-play for deployed systems. The basic idea is to detect AE attacks based on an intrinsic characteristic of AE, i.e., the instability of model prediction for AE, whose effectiveness has been validated in the image domain [65] and speech domain [22]. To realize such a detection method, we face several challenges. *AE algorithm variation:* For a detection system, the generating algorithm adopted by inputted AE is unknown, introducing the limitation that the detection needs to avoid involving any algorithm-specific design. *Adaptive setting:* The adversary may know all details of our detection system design and implementation, indicating that the detection needs to maintain effectiveness under the adaptive setting. *Architecture Uncertainty:* The guarded SR system may be built on different neural network architecture, introducing a critical demand for seamlessly transferring from one architecture to another architecture.

[*]Corresponding author

In this paper, we first illustrate the threat model of a targeted AE attack on SR systems. We summarize four design goals, including effectiveness, resilience, robustness and portability for a practical AE detection method. Considering the integrated audio distortion techniques as core components, their performance roughly decides the detection system's performance. Thus, we then take a feasibility study to evaluate a wide range of audio distortion techniques and select six alternative techniques to better satisfy our design goals. Encouraged by the study results, we propose *FraudWhistler*, which is resilient against various AE attacks, robust in the physical world and plug-and-play for any deployed SR system. *FraudWhistler* first generates several audio variants for the original audio sample with multi-channel audio distortion techniques. To observe the instability of the original audio sample, *FraudWhistler* extracts statistics from the outputs of the SR system for these audio variants. Based on that, *FraudWhistler* adopts the statistics as feature vectors, then employ a one-class SVM classifier to learn the decision boundary between adversarial examples and benign examples. Experimental results on different AE attacks, different physical conditions (e.g., device and communication channel), and adaptive attack setting show that *FraudWhistler* accurately detects adversarial examples, outperforming the state-of-the-art detection methods.

Our contributions are highlighted as follows:

- We propose an audio distortion-based AE detection method for speaker recognition, which is independent of AE-generating algorithms, effective even under adaptive attack settings, robust in complex realistic conditions, and plug-and-play for any deployed SR system.

- We present a study evaluating a wide range of audio distortion techniques from several aspects. The result is useful for the research community to make further progress on audio adversarial example detection.

- We design a multi-channel audio distortion method to generate audio variants combined with a feature extraction algorithm, which reveals information on the instability of model prediction for adversarial examples.

- We conduct extensive experiments on cutting-edge SR systems, employing five AE-generating algorithms across various physical environments. The results reveal that *FraudWhistler* attains an impressive overall accuracy of 98.7% on adversarial examples, with a minor degradation of 6.1% on benign samples. Also, *FraudWhistler* can achieve an accuracy of 84% even in the worst case against an adaptive adversary. Further, we demonstrate that *FraudWhistler* can achieve an average accuracy of 96.7% with a standard deviation of 1.73% across three different SR models.

The remainder of this paper is organized as follows. We introduce the threat model and design goals followed by the feasibility study in Section 2. In Section 3, we provide an in-depth explanation of the system design employed by *FraudWhistler*. Subsequently, Section 4 and Section 5 present the performance evaluation of *FraudWhistler* against the static adversary and adaptive adversary, respectively. Furthermore, we delve into a discussion of *FraudWhistler* in Section 6 and review relevant prior work in Section 7. Ultimately, we conclude our work in Section 8.

2 Preliminary

2.1 Background

Speaker Recognition. Speaker Recognition (SR) is an automatic technique that enables machines to recognize a speaker's identity from the voice characteristics. Benefiting from deep learning techniques, the SR systems currently are making progress rapidly in the past few years. SR systems have been supported by many open-source platforms (e.g., Kaldi [48] and Alize [32]) and integrated into various commercial products (e.g., Microsoft Azure, Apple Siri, Amazon Alexa, and Google Home). Recently, speaker recognition evaluation of NIST [40] has demonstrated that the latest SR systems are all based on deep learning (e.g., Ecapa-TDNN [15], SincNet [49] and ResNet34 [12]).

Adversarial Example. Goodfellow *et al.* [17] first demonstrated that the DNN models could misjudge an input (either image or audio) when an adversary adds some well-designed noise into it. These inputs with mathematically-designed perturbations are so-called adversarial examples. Let $F(\cdot)$ represent the DNN function, finding an adversarial example perturbation could be formulated as an optimization problem:

$$\begin{gathered} min\ \|\delta\|,\ s.t.\ F(X+\delta)=y_t,\ F(X)=y_s, \\ and\ y_t \neq y_s. \end{gathered} \tag{1}$$

where $\|\cdot\|$ represents the L norm, y_t denotes a target label for targeted attacks or any label but y_s for untargeted attacks. In the white-box setting, the adversary has full knowledge of the victim SR model (e.g., architecture and parameters). Instead, in the black-box setting, the adversary has nearly no knowledge of the victim SR model. While the adversarial example exposes the practical vulnerability of all deep learning-based applications, it is more threatening to SR systems that function as identity authentication.

2.2 Threat Model and Design Goals

As mentioned in Section 2.1, adversarial example attacks have practical threats to SR systems. It is necessary to provide a powerful detection to enable a secure and reliable SR system. In this section, we investigate the threat model and further define the design goals based on the model.

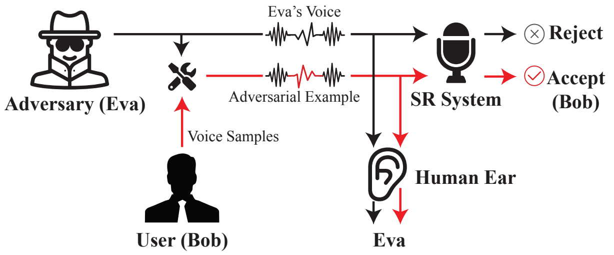


Figure 1: Threat model of adversarial example attack.

A Threat Model

An adversary aims to launch a targeted and imperceptible AE attack. As shown in Figure 1, the adversary (Eva) attempts to spoof an SR system to acquire legal identity (Bob) by adding specifically designed perturbation on her own voice. At the same time, she intends to keep the perturbed audio sample indistinguishable from the original audio, i.e., it still sounds like Eva's natural utterance to avoid bystanders' awareness. To generate the perturbation, Eva requires specific information about the SR system. Based on Eva's level of knowledge about the SR system, we consider two types of adversaries in our threat model. *(1) Static Adversary:* The adversary treats the system as unprotected and has complete knowledge of the SR model (e.g., architecture and parameters). Based on this knowledge, the adversary can reconstruct the same SR model and further obtain the prediction score, prediction label, and gradient from backpropagation. *(2) Adaptive Adversary:* The adversary knows not only the details of the SR system but also the detection method including architecture, parameters, and auxiliary data (e.g., noise source). Based on this knowledge, the adversary can perform attacks with adaptation to the detection. To cover most attack cases, we assume the adversary is with minimal restrictions. Specifically, the adversary cannot attack the training process (e.g., backdoor attacks and poisoning attacks [70]). Except that, the adversary is not restricted in any way to craft and hide the perturbation, as well as the AE attack launching (e.g., over-the-line and over-the-air).

B Design Goals

To make a practical detection that could be employed in realistic scenarios, we need to take both the adversary and the victim SR model into consideration. On the one hand, the adversary may launch AE attacks with different specific goals (e.g., universal attack and inaudible attack) through different communication channels (e.g., over-the-line and over-the-air). Thus, the detection needs to protect the victim SR model effectively in complex physical conditions. On the other hand, for a deployed SR system, retraining or modifying the system entails significant additional efforts, making the detection hard to deploy. Thus, the detection needs to avoid involving modifications to the SR system itself. To meet these considerations, the detection should satisfy the following design goals:

- **Effectiveness.** The detection should distinguish adversarial examples with high accuracy while remaining a low error rate for benign examples.
- **Resilience.** The detection should be resilient, i.e., cover as many as possible existing AE attacks and keep effective even under adaptive attacks [5].
- **Robustness.** The detection should remain robust in different communication channels or various physical environments (e.g., noise level and different reverb effects).
- **Portability.** The detection needs to be plug-and-play and function seamlessly with any existing deployed SR system without significant additional efforts.

2.3 Feasibility Study

According to the definition of adversarial examples [55], the subtle noise added to the original sample is an intrinsically elaborate perturbation. With this elaborate perturbation added, the adversarial example is at precise locations, i.e., the model predictions of AE are unstable to small changes. In contrast, the model predictions are stable for benign examples. Our basic idea is to utilize this different stability to detect AE.

To realize such a detection system, we have two challenges to settle: *1) What kind of techniques could be used to expose the dissimilar characteristics between benign examples and adversarial examples? 2) To achieve better detection performance, what else conditions should these techniques satisfy?* Existing works in speech domain [22,31] take pieces from the image domain and utilize some input transformation techniques to detect AE attacks. Such input transformation introduces an additional perturbation that can interfere with the carefully generated perturbation's function. According to this observation, we believe any technique that introduces distortion to audio samples could expose the dissimilar characteristics between benign and adversarial examples. Following this, we first explore all potential distortion methods as alternatives and evaluate them in many aspects to choose appropriate ones to further devise our detection.

A Alternative Methods

There are plenty of techniques that could distort audio samples. Basically, these techniques could be categorized into three types. The first type is traditional signal processing methods, e.g., resample and filtering. The second type is audio augmentation methods [16,19,21,26–28,41,42,47,50,67], e.g., add noise and add reverberation. The third type consists of more advanced audio reconstruction methods based on Linear Predictive Coding [45] and GriffinLim [18]. In Table 1, we list all involved distortion techniques and corresponding descriptions.

B Measurement Metrics

To select appropriate distortion methods, we need to measure all involved techniques according to our design goals. We first formulate our design goals as measurable metrics. In general, SR systems consist of a speaker embedding extracting module and a similarity prediction module. The similarity prediction results can be further utilized according to the specific task (e.g., speaker verification or speaker classification).

Basically, audio distortion is likely to impact the output of the SR system on the audio sample. More consistent outputs between original and distorted audio reflect that the model prediction on the original audio sample is more stable. Thus, we could roughly estimate the stability of model prediction on an audio sample by observing the consistency of system outputs between the original and distorted audio samples. To measure this consistency, we define a *Difference Score* function $DS(\cdot)$:

$$DS(x) = Sim(x,e) - Sim(d(x),e). \tag{2}$$

where x, e, $d(\cdot)$ and $Sim(\cdot)$ denote the benign example, the enrolled embedding, the distortion technique and the similarity score function, respectively. To better detect audio AE, the consistences revealed on benign and adversarial examples should be more distinguishable. To measure the distinguishing capability of a given distortion technique, we define a *Distinguishable Difference Score* function DDS_d :

$$DDS_d = DS(x') - DS(x). \tag{3}$$

where x' represents the targeted adversarial example. To measure this characteristic against an adaptive adversary, we define DDS'_d similar to DDS_d. To make practical detection, we measure the extra time cost of a distortion technique defined as follows:

$$ETC = \frac{T_{dist} - T_{base}}{T_{base}}. \tag{4}$$

where T_{base} and T_{dist} denote the running time without and with distortion, respectively.

C Analysis

Based on these metrics, we further conduct a feasibility study. We choose Ecapa-TDNN pre-trained on VoxCeleb as the speaker embedding extracting module, and Cosine-Similarity as the similarity function, so the output of $Sim(\cdot)$ ranges from -1.0 to 1.0. We employ one of the most common AE-generating algorithms (PGD [39]) to generate adversarial examples. In adaptive adversarial example attacks, we optimize a CW-like [6] objective function using PGD:

$$\begin{aligned} &min\ \mathcal{L}(SR(x+\delta),t) + \mathcal{L}(SR(d(x+\delta)),t) \\ &s.t.\ \|\delta\| < \varepsilon, \end{aligned} \tag{5}$$

where $\mathcal{L}$ represents CTC-Loss and t denotes the target speaker label. Since a differentiable implementation is not available for some distortion techniques (e.g., Quantization, GriffinLim, Codec and LPC), we use Backward Pass Differentiable Approximation (BPDA) technique [3] to replace direct gradient propagation. For those distortion techniques with randomness (e.g., Noisifier, DropChunk, DropFreq and Reverber), we use Expectation of Transformation (EOT) algorithm [4] to estimate the robust backward gradient.

For fair comparisons among these distortion techniques themselves, we align their attacking capacity with a fixed ε in

Table 1: Description and feasibility study results of each distortion technique. Underlined value means relatively bad performance, and each gray row has at least one underlined value.

Type	Distortion	Description	DDS_d	DDS'_d	ETC
Signal Processing	Quantization	Quantize each data point and then convert back by De-Quantization	0.61	0.24	≈ .001
	Codec	Compress audio sample and decompress	0.64	0.23	≈ .026
	Resample	Downsample the audio wave and upsample to original sample rate	0.34	-0.09	≈ .036
	Filtering	Filter the audio wave with high-pass and low-pass filters	0.18	-0.16	≈ .031
	Smoothing	Smooth the audio wave with specific window length	0.43	-0.20	≈ .007
Audio Augmentation	Noisifier	Add white noise with given SNR	0.68	0.59	≈ .006
	Reverber	Add reverberation effect with given RIR	0.47	0.27	≈ .113
	TimeScale	Scale the speed of audio	0.34	-0.18	≈ .039
	Clip	Clip the audio wave amplitude to certain range	0.07	-0.12	≈ .001
	DropChunk	Drop some chunks from audio wave	0.38	0.15	≈ .031
	DropFreq	Drop some frequency components from audio wave	0.39	0.18	≈ .224
	PitchShift	Shift the pitch level of the audio sample	0.42	-0.06	≈ 36.3
	TimeShift	Shift specific ratio of audio wave	0.11	-0.001	≈ .001
Audio Reconstruction	GriffinLim	Extract MelSpectrogram and reconstruct with GriffinLim [18]	0.40	0.48	≈ 295
	LPC	Extract LPC coefficients and reconstruct with random excitation	0.17	0.16	≈ 16.1

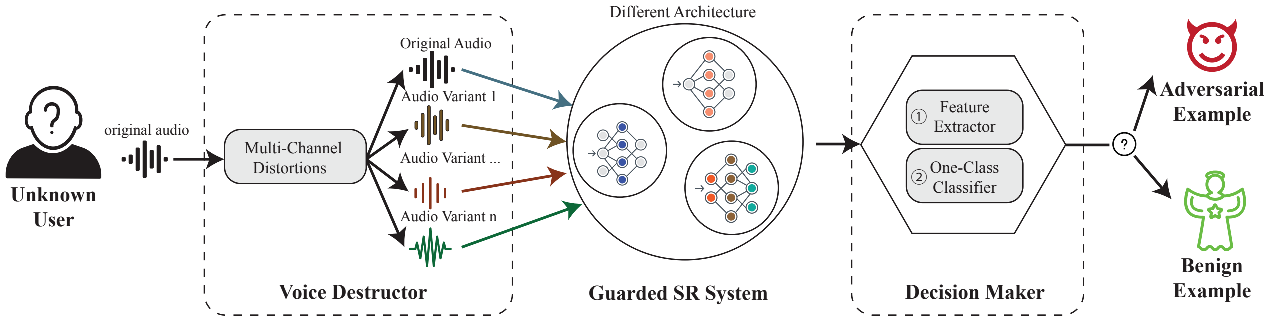


Figure 2: Framework of *FraudWhistler*.

PGD and optimize the parameter for each technique to achieve the highest DDS_d. The experimental results are shown in Table 1. Specifically, all distortion techniques except Clip could achieve DDS_d higher than 0.10. The highest DDS_d (0.68) is achieved by Noisifier and the lowest DDS_d is achieved by Clip. Every distortion technique suffers great degradation from white-box AE attacks (DDS_d) to adaptive AE attacks (DDS'_d) except Noisifier (only 0.09 degradation). It is noted that DDS'_d of some techniques decline below zero, indicating that for these techniques, $DS(x')$ is lower than $DS(x)$ under adaptive AE attacks. In these cases, the adaptive adversarial examples are more robust against audio distortions than benign examples. This also means these distortion techniques are vulnerable to adaptive AE attacks. Meanwhile, three techniques (PitchShift, GriffinLim and LPC) induce at least 16× computation time while other techniques introduce less than 0.2× computation time.

Revisiting our design goals, we have the following considerations. First, the technique should not cause unacceptable response delay. Second, the technique should keep effective under either a static adversary or an adaptive adversary. Combined with the aforementioned results, we choose six alternative distortion techniques (white rows in Table 1) and further devise our detection system based on them.

3 Methodology

In this section, we further present our detection framework and system design based on the threat model, design goals, and feasibility study.

3.1 Detection Framework

The goal of our detection scheme is to distinguish adversarial examples from benign examples. The basic idea is that the outputs of the SR system are unstable for adversarial examples while being stable for benign examples, as mentioned in Section 2.3.

Inspired by this, we propose *FraudWhistler* for detecting audio adversarial examples, as shown in Figure 2. When a user submits an audio sample to our system for identity verification, the system initially feeds the audio sample to the *Guarded SR System*, which generates a reference prediction result. Based on the reference prediction result, *FraudWhistler* needs to make a final decision on whether the audio sample is an adversarial example or a benign example.

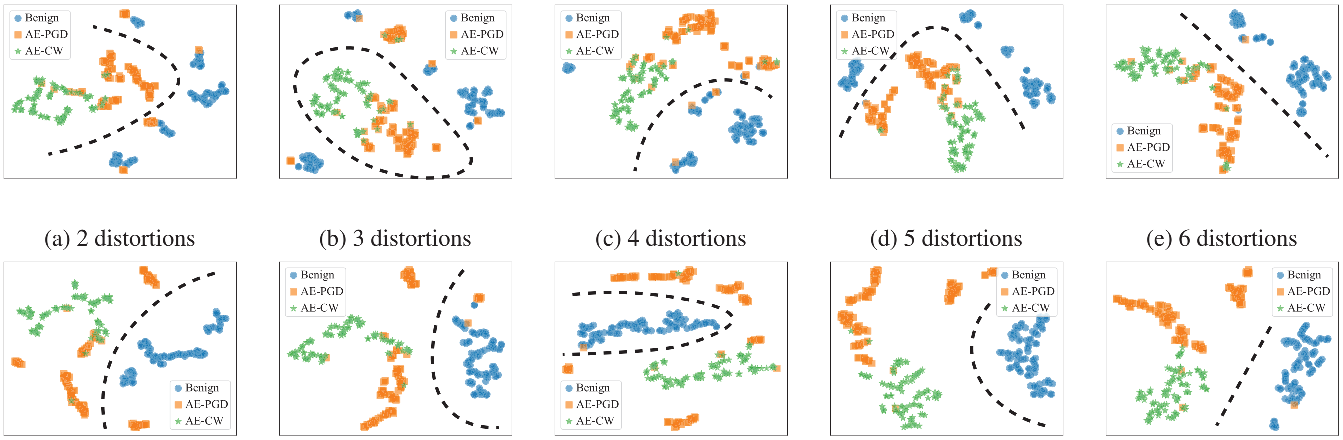
To achieve this, *FraudWhistler* employs a multi-step approach. Specifically, the *Voice Destructor* in *FraudWhistler* first apply various distortion techniques including Quantization, Codec, Noisifier, Reverber, DropChunk and DropFreq to generate audio variants. Based on the variants, the *Guarded SR System* generate corresponding prediction results. Combined with the reference prediction result, the *Decision Maker* in *FraudWhistler* extract a feature vector and feed it into a one-class SVM classifier to determine the audio sample is an adversarial example or not. If *FraudWhistler* predicts that the audio sample is a benign example, the user is identified as a legitimate user. Otherwise, the user is identified as an illegal user.

3.2 Voice Destructor

To differentiate adversarial examples based on the instability of SR system outputs, the *Voice Destructor* in *FraudWhistler* first generates several audio variants using a Multi-Channel distortion module.

As mentioned in Section 2.3, there are plenty of techniques that could be utilized to generate audio variants. There are two strategies to generate audio variants, i.e., different distortion techniques with fixed parameters or a specific distortion technique with various parameter settings. To explore the effectiveness of detecting adversarial examples for above strategies, we conduct a preliminary experiment. Specifically, we generate audio variants with a different number of distortion techniques and with various parameter settings (SNRs) of one specific distortion technique (Noisifier). In the latter strategy, we use the number of distortion levels to represent the number of parameter settings.

Figure 3 shows the visualization of classification performance by both strategies using t-SNE. By comparing the



(a) 2 distortions (b) 3 distortions (c) 4 distortions (d) 5 distortions (e) 6 distortions

(f) 2 distortion levels (g) 3 distortion levels (h) 4 distortion levels (i) 5 distortion levels (j) 6 distortion levels

Figure 3: t-SNE results for two distortion strategies, i.e., employ a different number of distortion techniques or a specific technique (Noisifier) with various distortion levels (SNRs).

result of Figures 3a to 3e, we can observe that audio variants generated from more distortion techniques contribute to creating more accurate and simple decision boundary, i.e., the number of incorrectly clustered samples decreases and the decision boundary is less complex. Similarly, the comparison between the results of Figures 3f to 3j demonstrates that more distortion levels for a specific distortion technique also improve the classification performance. Based on these two observations, the *Voice Destructor* deploys a Multi-Channel module consisting of six distortion techniques including Quantization, Codec, Noisifier, Reverber, DropChunk and DropFreq, in which Quantization and Noisifier are configured with multiple distortion levels.

3.3 Decision Maker

Combined with audio variants generated by *Voice Destructor* and the original audio sample, the *Guarded SR system* generates a sequence of prediction results, which includes one reference result. To predict whether the input audio sample is an adversarial example or not, *FraudWhistler* utilizes these prediction results to make a final decision.

Algorithm 1 Feature Extraction

```
Input:  score sequence $S$, reference score $s_{ref}$
Output:  feature vector $F$
Initialize $D$ as size of $S$
Initialize $Del$ as empty vector
for all $s \in S$ do
    $ds \leftarrow s - s_{ref}$
    $D \leftarrow D.append(ds)$
end for
$Stat \leftarrow [Var(D), Range(D), Mean(D), Max(D)]$
for $ds_1 \in D$ do
    for $ds_2 \in D - ds_1$ do
        if $ds_1, ds_2$ is from same distortion $d$ then
            $Del \leftarrow Del.append(abs(ds_1 - ds_2))$
        end if
    end for
end for
$F \leftarrow Concatenation(D, Del, Stat)$
return $F$
```

To begin with, we need to choose an appropriate classification model to match our demands. According to our design goals, we have the following two considerations. In principle, a simpler system induces fewer vulnerabilities and resource consumption, so we employ a support vector machine as our classifier to make our system more practical. Furthermore, to liberate our detection from the dependence on any pre-assumed AE-generating algorithms, we specifically utilize a one-class SVM classifier. To achieve accurate detection performance, we also need to devise an appropriate feature representation. Specifically, in *Decision Maker*, we extract a feature vector that signifies the instability of the original audio sample, leveraging the reference prediction result and prediction results obtained from the SR system on audio variants generated by *Voice Destructor*, as described in Algorithm 1. In detail, the reference prediction score is subtracted from every prediction score of audio variants, generating a sequence of ds values that represent the score differences. The sequence is notated as D and based on that, the variance, range, mean and max statistics are calculated to form a vector $Stat$. Additionally, for distortion techniques with multiple parameter settings, we compute the difference in ds values to construct a vector *Del*. Finally, the three vectors, namely D, *Del* and *Stat*, are concatenated together to create a comprehensive feature vector F, which is further fed into the SVM classifier.

4 Evaluation

In this section, we evaluate *FraudWhistler* against a static adversary in different SR architectures with large-scale datasets.

4.1 Experimental Setup

A Dataset

We implement *FraudWhistler* based on a large-scale corpus VCTK [66], which contains speech data uttered by 110 English speakers with various accents. Each speaker reads out about 400 sentences, which were selected from a newspaper. And each speaker has a different set of newspaper texts selected based on a greedy algorithm that increases the contextual and phonetic coverage. Among them, we randomly select 10 speakers (7 males and 3 females) as target users and for each target user, we select 10 utterances as enrolled utterances for SR systems. After removing utterances used for enrollment, we select 70 utterances for each target user as the benign examples for training the SVM classifier and 30 utterances for testing. To prepare the adversarial examples, we first select another 10 speakers for each target user from the remainder of the speakers available after removing the 10 target speakers. A target user and one of its adversary speakers together constitute an adversary trial. For each adversary trial, we randomly select an utterance from the adversary's utterances as the foundation for generating adversarial examples using five attack algorithms. In sum, we have 700 benign trials as the training dataset and 800 trials that include 300 benign trials and 500 adversarial trials as the testing dataset.

B Implementation

FraudWhistler is deployed on a server with 40 Intel Xeon Silver 4210R CPU, 256GB RAM, and four 48GB NVIDIA RTX A6000 GPU, running Ubuntu hirsute 21.04. In the *Voice Destructor* module, six distortion techniques are employed including noisifier, reverber, codec, quantization, dropchunk, and dropfreq. In noisifier, SNR is configured with 1 dB and 10 dB. In quantization, the parameter q is set to 7 and 8. The reverber employs Simulated Room Impulse Response (RIRs) [29]. The codec algorithm is flac, and each sample is stored in 8 bits. Within dropchunk, random dropping of 50 to 150 chunks occurs, with each chunk's length ranging from 100 to 1000 samples. In dropfreq, random dropping of 10 to 15 frequency chunks occurs, where each chunk has a range of 400 Hz. Regarding the one-class SVM classifier, we utilize the radial basis function (RBF) kernel.

Table 2: Notations used in metric definitions. AE represents Adversarial Examples, and BE represents Benign Examples. True (AE) represents the number of samples for which the true label is AE, and the same applies to the other notations.

Condition	True (BE)	True (AE)	
		Successful	Failed
Predicted (BE)	M	$P1$	$P2$
Predicted (AE)	N	$Q1$	$Q2$

C Guarded SR Systems

To evaluate the performance of *FraudWhistler*, we select the state-of-the-art SR system based on Ecapa-TDNN [15] with a Cosine-Similarity scorer. This model is pre-trained with another large-scale corpus VoxCeleb1 [44], which contains 1,251 speakers and 153,516 utterances. Based on this model, we build three SR systems for Automatic Speaker Verification (ASV), Close-set Speaker Identification (CSI) and Open-set Speaker Identification (OSI).

D Experiment Design

We evaluate *FraudWhistler* against a static adversary on three SR systems employing five AE attack methods in both the digital and physical world. In the digital world experiment, we directly use the dataset described before. For each AE attack method, we generate 100 audio adversarial examples for one given SR system. In the physical world experiment, we play the generated adversarial examples and record them in different physical conditions, varying the distance between the speaker and microphone, the background noise level, and the recording device. In sum, we evaluate *FraudWhistler* on 1500 AE attacks in the digital world and 3000 AE attacks in the physical world.

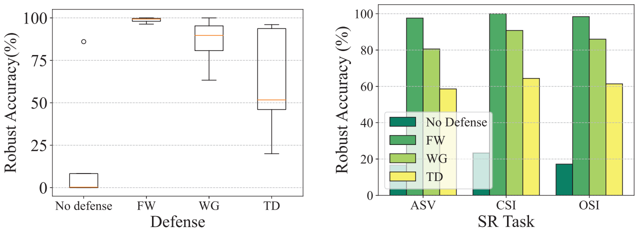
E Evaluation Metrics

For evaluations, we define the following metrics with the notations described in Table 2:

- Detect Accuracy on AEs (ACC_{ae}): $ACC_{ae} = \frac{Q1+Q2}{P1+Q1+P2+Q2}$, which is the detection rate on adversarial examples.

- Accuracy on BEs (ACC_{be}): $ACC_{be} = \frac{M}{M+N}$, which reflects the impact on benign examples.

- Robust Accuracy on AEs (ACC_{rob}): $ACC_{rob} = 1 - \frac{P1}{P1+Q1+P2+Q2}$, which stands for the whole system's robustness against AE attacks.



(a) Different algorithms

(b) Different Architectures

Figure 4: Performance of *FraudWhistler* across various AE-generating algorithms and different SR architectures.

4.2 Overall Performance

We first evaluate the overall performance of *FraudWhistler* in terms of effectiveness against the static adversary. Table 3 shows the evaluation results of *FraudWhistler* (FW) and two state-of-the-art (SOTA) works WaveGuard (WG) [22] and TemporalDependency (TD) [69]. We can observe that *FraudWhistler* achieves over 40% ACC_{ae} improvement compared with TD in all SR architectures while inducing only 0.3% ACC_{be} degradation. Also, *FraudWhistler* outperforms WaveGuard with at least 7% ACC_{ae} improvement while inducing about 2% ACC_{be} degradation. With *FraudWhistler* deployed, SR systems achieve 97.6% ACC_{rob} at worst, compared to 80.6% for WaveGuard and 58.6% for TD, indicating strong defense capability against adversarial examples.

4.3 Evaluation on Resilience

As described in our threat model, the adversary is not restricted in any way on crafting and hiding the adversarial perturbations, indicating that the adversary could use any AE-generating algorithms. We take several classic algorithms in AE attack domain including FGSM [17], PGD [39] and CW [6] and two more advanced algorithms, FM [60] and UNIV [33, 37, 64, 73]. FM makes use of the psychoacoustic principle of frequency masking to generate inaudible adversarial examples, and UNIV is designated to generate universal adversarial perturbation that can be added to any speaker's speech.

Figure 4a shows the performance of *FraudWhistler* on different algorithms, we can observe that *FraudWhistler* achieves 98.7% ACC_{rob} at average, compared to 85.5% for WG and 61.5% for TD, respectively. Also, it can be observed that *FraudWhistler* achieves steady performance with the range of 3.7%, compared to 36.7% for WG and 76% for TD, respectively. Further analysis shows that, though WG achieves over 90% ACC_{rob} for FGSM, PGD, and CW, its accuracy degrades to lower than 80% for FM and UNIV. Meanwhile, TD achieves over 90% ACC_{rob} only for FGSM and CW, and its accuracy degrades to lower than 50% for PGD, FM, and UNIV.

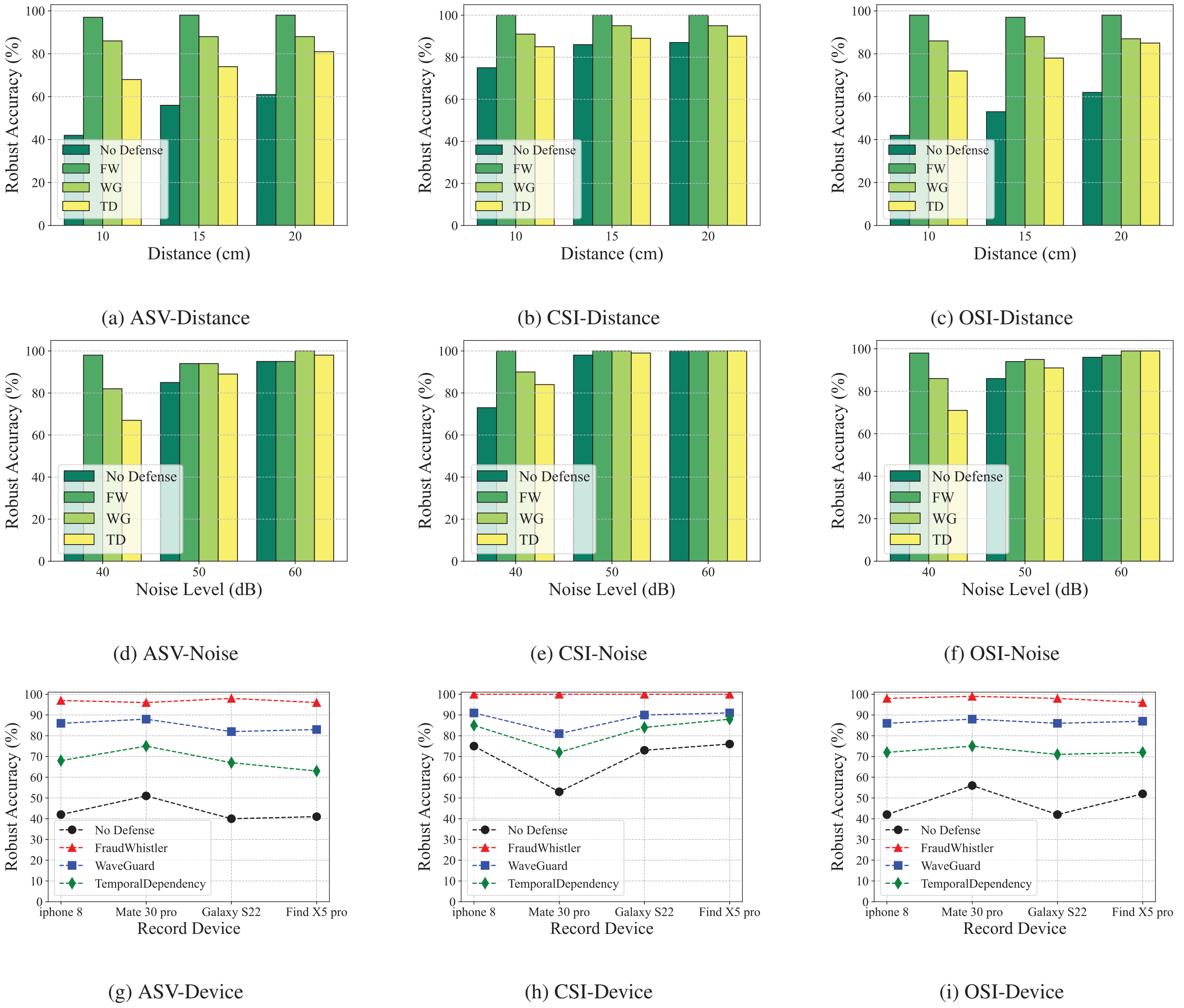
To investigate the impact of SR architectures on the detection's performance, we further evaluate *FraudWhistler* on different SR architectures. Figure 4b shows that *FraudWhistler* achieves 97.6% ACC_{rob} at worst on three SR architectures, compared to 80.6% for WG and 58.6% for TD. Also, it can be observed that these three detection methods achieve the best performance on CSI and the worst performance on ASV. This result demonstrates that *FraudWhistler* achieves steadily high accuracy on adversarial examples across different AE generating algorithms and SR architectures, which validates its resilience in a practical scenario for deployed SR systems to defend against AE attacks.

4.4 Evaluation on Robustness

In a practical attack scenario, the adversary launch attacks in complex physical environments. Hence, we further evaluate *FraudWhistler* in the physical world. To simulate practical attacks, we employ a speaker (JBL Clip3) to play audio adversarial examples and use smartphones to record audio, as illustrated in Figure 6. The speaker and smart devices are placed on a 1.5 $m \times 3.6$ m table, in a 6 $m \times 7$ m room. We take three variables to control the physical environment condition, i.e., the distance between the speaker and the microphone of the smart device, background noise level and type of smart device. To control the background noise level, we place another smart device to play white noise and measure the noise level at the location of the recording devices' microphone, with a digital sound level meter (Smart Sensor AR844).

Table 3: Overall performance of *FraudWhistler* and SOTA works against the static adversary.

SR Architectures	$ACC_{ae}(\%)$			$ACC_{be}(\%)$				$ACC_{rob}(\%)$			
	FW	WG	TD	No Defense	FW	WG	TD	No Defense	FW	WG	TD
ASV	**87.8**	80.6	43.6	99.67	94.33	92.00	**94.67**	16.6	**97.6**	80.6	58.6
CSI	**86.2**	69.0	45.6	100.0	94.33	**96.00**	92.00	23.4	**100**	90.8	64.4
OSI	**94.0**	86.0	49.6	100.0	92.67	**94.67**	93.33	17.2	**98.4**	86.0	61.4



(a) ASV-Distance

(b) CSI-Distance

(c) OSI-Distance

(d) ASV-Noise

(e) CSI-Noise

(f) OSI-Noise

(g) ASV-Device

(h) CSI-Device

(i) OSI-Device

Figure 5: Performance of *FraudWhistler* and SOTA works in the physical world varying on the distance between the speaker and microphone, the background noise level or the recording devices.

Figures 5a to 5c show the performance of *FraudWhistler* varying the distance between the speaker and microphone of the recording device. We can observe that *FraudWhistler* achieves best ACC_{rob} (97% at worst) in all distance settings, compared to WG and TD. Also, it can be observed that as the distance increases, the SR system itself without defense and TD both achieve higher ACC_{rob}. Considering different SR architectures, we can observe *FraudWhistler* achieves steady performance, while WG and TD achieve higher accuracy for CSI than ASV or OSI. Figures 5d to 5f show the performance of *FraudWhistler* varying the background noise level. We can observe that when the environment is relatively quiet (e.g., with the background noise level as 40 dB), *FraudWhistler* achieves the best accuracy (above 98%) for all SR architectures. However, as the background is louder, the SR system without defense achieves 45%, 25%, and 44% ACC_{rob} improvement for ASV, CSI, and OSI respectively. When the background noise level reaches 60 dB, the SR system itself achieves at worst 95% ACC_{rob} across three architectures. Figures 5g to 5i show the performance of *FraudWhistler* and SOTA works with various record devices. We can observe that *FraudWhistler* achieves above 96% ACC_{rob} in all device settings. Also, it can be observed that *FraudWhistler* has at most 3% ACC_{rob} variation, while WG and TD have 10% and 14% variation, respectively. This result demonstrates that *FraudWhistler* could detect adversarial examples with steady

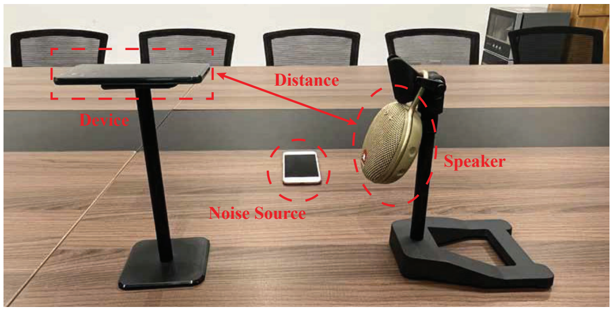


Figure 6: Physical setting to simulate practical attacks.

and accurate performance in various environmental conditions, which indicates its robustness in the complex physical world.

4.5 Evaluation on Transferability

As described in Section 3, our approach is devised to be plug-and-play for deployed SR systems whose neural network models vary in real-world scenarios. In real-world scenarios, during the training phase, *FraudWhistler* relies on an SR system for training the SVM classifier. However, *FraudWhistler* may be applied to another deployed SR system, which may have a different neural network model compared to the one used during training. Hence, it is necessary to evaluate the transferability of *FraudWhistler* across different SR models. Specifically, we evaluate *FraudWhistler* with three SR systems including Ecapa-TDNN [15], X-vector [57] and GE2E [58] which are the state-of-the-art SR model, the most classic model, and representative real-world deployed model (e.g., applied in Resemblyzer [1]), respectively. To keep consistent with a real-world deploying procedure, we did not conduct any fine-tuning operations in this experiment.

Figure 7 shows the robust accuracy (ACC_{rob}) of *FraudWhistler* across three SR models. To distinguish between the SR systems used in training and inference, we refer to them as Train-SR and Test-SR, respectively. We can observe that *FraudWhistler* trained with Ecapa-TDNN achieves the best performance across three Test-SR systems, whose accuracies are all above 94%. Instead, *FraudWhistler* trained with GE2E achieves the worst performance across three Test-SR systems, whose accuracies are all below 90%. The lowest accuracy is achieved with the Train-SR and Test-SR as GE2E and X-vector, respectively. This is because the capability of depicting a speaker's characteristics in GE2E's speaker embedding space is less powerful. This result demonstrates that *FraudWhistler* has excellent generalization capabilities when paired with an advanced SR model during the training phase, which is especially on benefits from the continuously evolving SR systems.

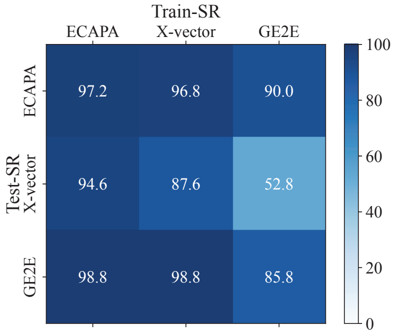


Figure 7: Performance of *FraudWhistler* when transferring among Ecapa-TDNN, X-vector and GE2E SR systems.

4.6 Performance Overhead

With *FraudWhistler* deployed, the SR systems may consume more computation resources, which induces a longer system response time and more memory usage. Hence, we evaluate the performance overhead brought by *FraudWhistler*. Table 4 shows the average Wall-Clock time and the memory usage of the SR system and *FraudWhistler*, respectively. We can observe that the average Wall-Clock time for Ecapa-TDNN SR is 178.6 ms and that of *FraudWhistler* is only 3.428 ms, indicating that *FraudWhistler* introduces approximately 1.92% additional running time only. We also evaluate the memory usage for the SR system and *FraudWhistler*. The result shows that the SR system occupies 84.647 MB memory and *FraudWhistler* occupies 2.324 KB memory, indicating that *FraudWhistler* introduces about 0.003% additional memory usage only, which is subtle compared to the original SR system. In sum, *FraudWhistler* introduces negligible performance overhead, enabling it to be a plug-and-play defense system in real-world scenarios.

4.7 Ablation Study

In this section, we investigate the impact of key components in *FraudWhistler* including Multi-Channel Distortions and Statistic Extractor. To this end, we implement two *FraudWhistler* variants FW-M and FW-S. In FW-M, we exclude both Multi-Channel Distortions and Statistic Extractor design,

Table 4: Performance overhead of *FraudWhistler*.

Process	Wall-Clock Time(ms)	Memory
Ecapa-TDNN SR	178.6	84.647MB
FraudWhistler	3.428	2.324KB
VoiceDestructor	3.160	-
DecisionMaker	0.268	-

i.e., generate audio variants in Voice Destructor using 6 distortion techniques with fixed parameter settings, and we directly use score sequences described in Algorithm 1 as a feature vector, without extracting statistics. In FW-S, we only exclude Statistic Extractor design, i.e., generate audio variants using 6 distortion techniques with various parameter settings and do not extract statistics. Besides, we also implement a variant named FW-OP with the setting that SR systems provide only predictions without similarity scores for effectiveness evaluation. In FW-OP, we set the value of ds as 1 if the predictions of the SR system for the original audio and the distorted one are different, otherwise set ds as 0, and other components are the same as in FW.

Table 5 shows the effectiveness of FW-M, FW-S, FW and FW-OP in three architectures. Comparing the first three systems, we can observe for ASV and OSI, FW achieves the best ACC_{rob} performance (98.2% at worst) while inducing 0.22% ACC_{be} degradation compared to FW-M. This result validates the effectiveness of Multi-Channel Distortions and Statistic Extractor in *FraudWhistler*. Comparing FW with FW-OP, it can be observed that with only prediction, FW-OP achieves lower accuracy in nearly all situations. The only exception is that in ASV task, it achieves higher ACC_{ae} by about 10% than FW, but at the same time, ACC_{rob} decreases to 31.11% which is unacceptable. This result validates the significance of the combination of scores and predictions in *FraudWhistler*.

5 Adaptive Attack

To enable reliable SR systems, it is important to evaluate *FraudWhistler* against the adaptive adversary. As described in Section 2.2, an adaptive adversary knows the details of the SR system, and is aware of the detection mechanism including architecture, parameters, and auxiliary data. In this section, we build an adaptive adversary and evaluate *FraudWhistler* against it.

Table 5: Performance of FW-M, FW-S, FW and FW-OP on different SR architectures.

SR Arch	Metric(%)	FW-M	FW-S	FW	FW-OP
ASV	ACC_{ae}	85.00	83.00	87.20	**98.2**
	ACC_{rob}	96.20	96.00	**98.20**	**98.2**
	ACC_{be}	92.33	93.89	**94.11**	31.11
CSI	ACC_{ae}	**88.40**	87.69	84.40	83.6
	ACC_{rob}	**100.0**	**100.0**	**100.0**	**100**
	ACC_{be}	**96.11**	94.11	95.89	87.22
OSI	ACC_{ae}	86.60	86.60	**94.60**	92.6
	ACC_{rob}	96.20	96.00	**98.40**	95.8
	ACC_{be}	92.22	93.44	**94.33**	90.89

5.1 Experimental Setup

In this experiment, we use the same dataset VCTK and Ecapa-TDNN SR system as in Section 4. To evaluate *FraudWhistler* under adaptive setting, we redefine ACC_{rob} and introduce new metrics:

- Adaptive Attack Success Rate (ASR): $ASR = \frac{N}{M}$, where N is the number of successful adversarial examples and M is the total number of attack trials.

- Robust Accuracy on adaptive AEs (ACC_{rob}): $ACC_{rob} = 1 - ASR$, which reflects the robustness of the guarded SR system against the adaptive adversary.

- Signal-to-Noise Ratio (SNR): $SNR = 10log_{10}(\frac{P_x}{P_\delta})$, where P_x and P_δ are the signal power of the original audio sample and the corresponding adversarial perturbation, respectively.

- Human Accuracy on adaptive AEs (ACC_{man}): ACC_{man} is the detection accuracy of humans on adaptive adversarial examples.

5.2 Attack Design

In this section, we introduce an adaptive AE-generating algorithm to bypass *FraudWhistler*. As motivated in Section 2.3, *FraudWhistler* detect AEs based on the model output of the AE is unstable under audio distortions. Thus, to bypass *FraudWhistler*, the adversary needs to craft the perturbation such that $SR(d(x+\delta))$ is stable when d is substituted with different distortion techniques. Meanwhile, considering the original objective to attack the SR system, the outputs of $SR(d(x+\delta))$ need to match $SR(x+\delta)$ closely. Therefore, to craft such a perturbation δ, the adversary needs to optimize a CW-like objective function as follows:

$$\begin{aligned} & min\ \mathcal{L}(SR(x+\delta),t)+\sum_{i=1}^{6} c_i \cdot \mathcal{L}(SR(d_i(x+\delta)),t) \\ & s.t.\ \|\delta\|_\infty < \varepsilon, \end{aligned} \tag{6}$$

where $\mathcal{L}$ represents CTC-loss, *epsilon* denotes perturbation budget, and c_i is the hyper-parameter to control the weights of respective loss. Considering six distortion techniques, quantization is non-differentiable, and four techniques (Noisifier, Reverber, DropChunk and DropFreq) are with randomness. Thus, we use BPDA [3] technique to replace direct gradient propagation for quantization, and use EOT [4] algorithm to estimate robust backward gradient for the aforementioned four techniques. In our implementation, we use Project Gradient Descent (PGD [39]) to optimize the objective function. The detailed algorithm is described in Appendix A.

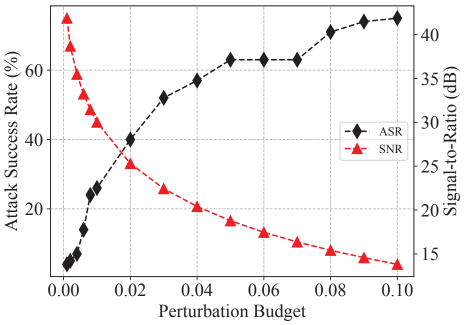


Figure 8: The attack success rate (ASR) and Signal-to-Noise Ratio (SNR) of adaptive attacks on *FraudWhistler*, while varying the perturbation budget.

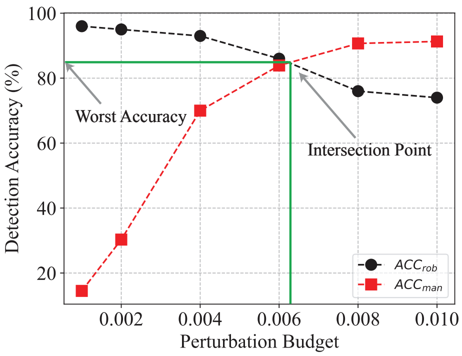


Figure 9: The detection accuracy of *FraudWhistler* and human ear on adaptive adversarial examples, while varying the perturbation budget.

5.3 Performance Evaluation

We conduct our adaptive attack evaluation in two aspects: (1) the attack success rate (ASR) and (2) the imperceptibility of adversarial examples. Toward this end, we vary the perturbation budget ε in our adaptive attacks and explore how these two sides change simultaneously. Specifically, we implement the proposed adaptive attack on three SR architectures and vary the perturbation budget from 0.001 to 0.10. The result is shown in Figure 8. Although we mentioned in Section 2.3 that these distortions could defend adaptive attacks with limited capability, i.e., a fixed parameter ε, we can observe that as the perturbation budget increases, the attack success rate could still reach higher. However, as the success rate increases, the SNR decreases. When ASR reaches approximately 75%, the SNR drops below 15 dB. This outcome suggests that *FraudWhistler* may be susceptible to adaptive attacks when the perturbation budget is relatively high. Nevertheless, maintaining the imperceptibility of adversarial examples is crucial in practical attack scenarios [11,37]. For this reason, we conduct another audibility study to evaluate *FraudWhistler*.

5.4 Audibility Study

Note that IRB approval is obtained in terms of our work involving human participants for audibility studies. In our audibility study, we recruit 31 volunteers (21 males and 10 females). We recruit volunteers on our campus forums. The recruited volunteers consist of undergraduate students, graduate students, and some faculties. We screen for hearing issues in our recruitment notice and the study was completed in the volunteers' environments. Before the start of the experiment, we prepared an informed consent form that outlines the research purpose, experimental procedures, and the usage of data. Participants may voluntarily choose whether or not to grant authorization for our audibility study. The minimum and maximum age of volunteers is 9 and 49, respectively, and most of them (24 people) are at the age of 18 to 25. The study took each volunteer about 15 minutes and every participant is compensated with $5. For each subject, we play 60 audio adversarial examples while keeping the volume to a normal level (about 60-70 dB) and let them identify whether the played audio is a benign example or an adversarial example. These 60 audio examples are randomly selected with a perturbation budget between 0.001 and 0.01.

As shown in Figure 9, we can observe that as the perturbation budget increases, the human ear could identify adversarial examples more accurately, while *FraudWhistler* achieves worse accuracy. This indicates that, though the adaptive adversary could improve the attack success rate with higher perturbation, the impact on imperceptibility would render adversarial examples failed to spoof human ears. The intersection point indicates that *FraudWhistler* and the human ear achieve the same accuracy. And when the perturbation budget is lower, *FraudWhistler* could detect adversarial more accurately. On the other hand, when the perturbation budget is higher than at this intersection, the human ear could easily identify adversarial examples. At the intersection point, the SR system achieves the worst accuracy (84%) on adversarial examples. This result indicates that *FraudWhistler* could force the adversarial examples to either fail to bypass the SR system or become noticeable to human ears, which enables a reliable SR system in the real world.

6 Discussion

In this section, we discuss the scalability of *FraudWhistler*, and possible advanced detection.

Detection Scalability: Backdoor attack [70] is another type of security threat for SR systems. We notice that in backdoor attacks, the adversary needs to contain a trigger in the audio sample which could also be perturbed by distortion techniques. Though backdoor attacks mainly focus on the training stage of SR systems, the adversary still needs to trigger the backdoor during inference time. This indicates that *FraudWhistler* may have the scalability to defend against backdoor attacks.

Possible Advanced Detections. To improve the security of the SR system, one can make possible advanced detection by modifying *FraudWhistler*. The experimental result has shown that the detection accuracy of *FraudWhistler* against static adversaries is satisfactory. On the other hand, *FraudWhistler* achieves a little lower accuracy against adaptive adversaries. One obvious strategy for better security is to introduce randomness in the multi-channel distortion component. We can use cryptographic randomness to make the distortion technique unpredictable, which complicates the task for adversaries. Further, based on the results of our transferability study, pairing the defense with a more advanced SR system during the training phase could potentially make up a more powerful defense.

7 Related Work

In this section, we discuss several key related work on adversarial example attacks over existing SR systems and corresponding defenses in the audio domain.

Adversarial Example Attacks on Speaker Recognition. We summarize existing adversarial example attacks on SR and find most of the works do not take defense into their consideration and are under white-box settings. The detailed summary is in Appendix B.

Kreuk *et al.* [30] implemented the first adversarial example attacks [55] on an end-to-end SR model [20] using *fast gradient sign method* (FGSM) [17]. Then, Li *et al.* [35] launched adversarial example attacks on GMM i-vector [14] systems and x-vector [54] systems. Villalba *et al.* [57] benchmarks adversarial examples' robustness in x-vector SR systems. The following works focus on launching practical adversarial attacks. These works [7,36,37,64,73] are mainly designated to generate robust adversarial examples to launch over-the-air attacks considering the room impulse response (RIR).

Another line of works [33, 37, 64, 73] is designated to generate universal adversarial examples that can be added to any specific speaker's speech. At almost the same time, some works intend to generate imperceptible perturbations. FoolHD [53] proposed steganography-inspired adversarial attacks to generate highly imperceptible perturbations. Wang *et al.* [60] proposed to generate inaudible adversarial perturbations based on the psychoacoustic principle of frequency masking. VoiceCloak [10] modulates perturbations into RIR. While most of the aforementioned works are under the white-box setting, FakeBob [7] conducts the first comprehensive and systematic study of the adversarial example attacks in the practical black-box setting using NES [23].

Defenses Against Adversarial Example in Audio Domain. We summarize existing defenses against adversarial example attacks on the SR model and find they can be categorized into three types. The detailed summary is in Appendix C.

The first line of works [46, 59] focuses on making SR systems more robust by retraining model with the knowledge of pre-assumed AE-generating algorithms. Wang *et al.* [59] use FGSM and *local distributional smoothness* [43] for model regularization while Pal *et al.* [46] use hybrid adversarial training considering FGSM, PGD [39], and CW [6]. The second line of works [25, 62, 71] are designated to extract benign parts from the perturbed audio example. To purify audio samples Zhang *et al.* [71] trained an adversarial separation network, Joshi *et al.* [25] proposed several preprocessing methods [13, 24, 39, 52, 68] as filtering module, and Wu *et al.* [62] use cascaded self-supervised learning models [38] to purify the adversarial perturbations. The third line of works [34, 61, 63] focuses on detecting AEs. Li *et al.* [34] introduced a VGG-like [44] binary detector while Wu *et al.* [61] adopt neural vocoders [68] to spot adversarial samples. Wu *et al.* [63] proposed a voting scheme by employing random sampling to detect AEs. Adversarial training-based and purification-based works either need extra efforts to retrain models or cause some negative effects on benign examples. The most related work [25] takes adaptive attack [56] and universal attack into consideration. Besides, almost every aforementioned work has a limited scope of attack algorithms and only one of them can resist unseen attacks successfully. Also, none of aforementioned work takes the deploying procedure in real-world scenarios into consideration.

Compared to these defenses, our work has a wide coverage of existing adversarial example attacks in both the digital world and the complex physical world. Besides, our work can defend the adaptive attacks by pushing the adversarial example either to be failed or perceptible Further, our defense can transfer well across different SR models, enabling it as a plug-and-play method in real-world scenarios. Also, our defense does not entail significant additional efforts to function seamlessly with deployed SR systems. All these features make our detection framework a practical defense for speaker recognition.

8 Conclusion

In this paper, we propose an audio distortion-based adversarial example detection method for the SR system and implement a practical detection system *FraudWhistler*. We present a study on different audio distortion techniques, and the result is helpful for the research community in developing more advanced adversarial example detection methods. Encouraged by the study result, we introduce a multi-channel distortion technique to generate audio variants. Based on the outputs of the SR system for these audio variants, we introduce an algorithm to extract features that represent the instability of the original audio sample. Further, we employ a one-class SVM classifier to learn the decision boundary between adversarial examples and benign examples. Experimental results in both the digital world and the physical world show that *FraudWhistler* can detect audio adversarial examples with steadily accurate performance and high efficiency.

Acknowledgements

We appreciate the shepherd's efforts and all the reviewers' valuable comments. This research is sponsored by National Natural Science Foundation of China (62102354, 62032021, 62172359, 61972348), Fundamental Research Funds for the Central Universities (2021FZZX001-27).

References

[1] Resemblyzer. https://github.com/resemble-ai/Resemblyzer, 2020.

[2] Hadi Abdullah, Kevin Warren, Vincent Bindschaedler, Nicolas Papernot, and Patrick Traynor. Sok: The faults in our asrs: An overview of attacks against automatic speech recognition and speaker identification systems. In *Proceedings of IEEE S&P*, pages 730–747, San Francisco, CA, USA, 2021.

[3] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of ACM ICML*, pages 274–283, Stockholm, Sweden, 2018.

[4] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *Proceedings of ACM ICML*, pages 284–293, Stockholm, Sweden, 2018.

[5] Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. *CoRR*, 2019.

[6] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *Proceedings of IEEE S&P*, pages 39–57, San Jose, CA, USA, 2017.

[7] Guangke Chen, Sen Chenb, Lingling Fan, Xiaoning Du, Zhe Zhao, Fu Song, and Yang Liu. Who is Real Bob? Adversarial Attacks on Speaker Recognition Systems. In *Proceedings of IEEE S&P*, pages 694–711, San Francisco, CA, USA, 2021.

[8] Guangke Chen, Zhe Zhao, Fu Song, Sen Chen, Lingling Fan, Feng Wang, and Jiashui Wang. Towards understanding and mitigating audio adversarial examples for speaker recognition. *IEEE Transactions on Dependable and Secure Computing*, pages 1–17, 2022.

[9] Meng Chen, Li Lu, Zhongjie Ba, and Kui Ren. Phoneytalker: An out-of-the-box toolkit for adversarial example attack on speaker recognition. In *Proceedings of IEEE INFOCOM*, pages 1419–1428, London, United Kingdom, 2022.

[10] Meng Chen, Li Lu, Junhao Wang, Jiadi Yu, Yingying Chen, Zhibo Wang, Zhongjie Ba, Feng Lin, and Kui Ren. Voicecloak: Adversarial example enabled voice de-identification with balanced privacy and utility. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 7(2), 2023.

[11] Qianniu Chen, Meng Chen, Li Lu, Jiadi Yu, Yingying Chen, Zhibo Wang, Zhongjie Ba, Feng Lin, and Kui Ren. Push the limit of adversarial example attack on speaker recognition in physical domain. In *Proceedings of ACM SenSys*, Boston, MA, USA, 2022.

[12] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. Voxceleb2: Deep speaker recognition. *arXiv preprint arXiv:1806.05622*, 2018.

[13] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *Proceedings of ACM ICML*, pages 1310–1320, Long Beach, California, USA, 2019.

[14] Najim Dehak, Patrick J. Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet. Front-end factor analysis for speaker verification. *IEEE ACM Trans. Audio Speech Lang. Process.*, 19(4):788–798, 2011.

[15] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck. Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification. *arXiv preprint arXiv:2005.07143*, 2020.

[16] Chenpeng Du and Kai Yu. Speaker augmentation for low resource speech recognition. In *Proceedings of IEEE ICASSP*, pages 7719–7723, Barcelona, Spain, 2020.

[17] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *Proceedings of ICLR*, San Diego, CA, USA, 2015.

[18] Daniel Griffin and Jae Lim. Signal estimation from modified short-time fourier transform. *IEEE Trans. Acoust. Speech Signal Process.*, 32(2):236–243, 1984.

[19] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, and Adam Coates. Deep speech: Scaling up end-to-end speech recognition. *CoRR*, 2014.

[20] Georg Heigold, Ignacio Moreno, Samy Bengio, and Noam Shazeer. End-to-end text-dependent speaker verification. In *Proceedings of IEEE ICASSP*, pages 5115–5119, Shanghai, China, 2016.

[21] Chien-Lin Huang. Exploring Effective Data Augmentation with TDNN-LSTM Neural Network Embedding

for Speaker Recognition. In *Proceedings of IEEE ASRU*, pages 291–295, SG, Singapore, 2019.

[22] Shehzeen Hussain, Paarth Neekhara, Shlomo Dubnov, and Julian McAuley. WaveGuard: Understanding and Mitigating Audio Adversarial Examples. In *Proceedings of USENIX Security*, pages 2273–2290, Virtual Event, 2021.

[23] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *Proceedings of ACM ICML*, pages 2137–2146, Stockholm, Sweden, 2018.

[24] Daniel Im Im, Sungjin Ahn, Roland Memisevic, and Yoshua Bengio. Denoising criterion for variational auto-encoding framework. In *Proceedings of AAAI AAAI*, San Francisco, California, USA, 2017.

[25] Sonal Joshi, Jesús Villalba, Piotr Żelasko, Laureano Moro-Velázquez, and Najim Dehak. Study of Pre-Processing Defenses Against Adversarial Attacks on State-of-the-Art Speaker Recognition Systems. *IEEE Trans. Inf. Forensics Secur.*, 16:4811–4826, 2021.

[26] Naoyuki Kanda, Ryu Takeda, and Yasunari Obuchi. Elastic spectral distortion for low resource speech recognition with deep neural networks. In *Proceedings of IEEE ASRU*, pages 309–314, Olomouc, Czech Republic, 2013.

[27] Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. Audio augmentation for speech recognition. In *Proceedings of ISCA INTERSPEECH*, pages 3586–3589, Dresden, Germany, 2015.

[28] Tom Ko, Vijayaditya Peddinti, Daniel Povey, Michael L. Seltzer, and Sanjeev Khudanpur. A study on data augmentation of reverberant speech for robust speech recognition. In *Proceedings of IEEE ICASSP*, pages 5220–5224, New Orleans, LA, 2017.

[29] Matěj Korvas, Ondřej Plátek, Ondřej Dušek, Lukáš Žilka, and Filip Jurčíček. Free English and Czech telephone speech corpus shared under the CC-BY-SA 3.0 license. In *Proceedings of the Eigth International Conference on Language Resources and Evaluation (LREC 2014)*, 2014.

[30] Felix Kreuk, Yossi Adi, Moustapha Cisse, and Joseph Keshet. Fooling end-to-end speaker verification with adversarial examples. In *Proceedings of IEEE ICASSP*, pages 1962–1966, Calgary, AB, Canada, 2018.

[31] Hyun Kwon, Hyunsoo Yoon, and Ki-Woong Park. Poster: Detecting audio adversarial example through audio modification. In *Proceedings of ACM SIGSAC CCS*, page 2521–2523, New York, NY, USA, 2019.

[32] Anthony Larcher, Jean-Francois Bonastre, Benoit Fauve, Kong Aik Lee, Christophe Levy, Haizhou Li, John S D Mason, and Jean-Yves Parfait. ALIZE 3.0 - Open Source Toolkit for State-of-the-Art Speaker Recognition. In *Proceedings of ISCA INTERSPEECH*, page 6, Lyon, France, 2013.

[33] Jiguo Li, Xinfeng Zhang, Chuanmin Jia, Jizheng Xu, Li Zhang, Yue Wang, Siwei Ma, and Wen Gao. Universal adversarial perturbations generative network for speaker recognition. In *Proceedings of IEEE ICME*, pages 1–6, London, UK, 2020.

[34] Xu Li, Na Li, Jinghua Zhong, Xixin Wu, Xunying Liu, Dan Su, Dong Yu, and Helen Meng. Investigating robustness of adversarial samples detection for automatic speaker verification. In *Proceedings of ISCA INTERSPEECH*, pages 1540–1544, Virtual Event, Shanghai, China, 2020.

[35] Xu Li, Jinghua Zhong, Xixin Wu, Jianwei Yu, Xunying Liu, and Helen Meng. Adversarial attacks on GMM i-vector based speaker verification systems. In *Proceedings of IEEE ICASSP*, pages 6579–6583, Barcelona, Spain, 2020.

[36] Zhuohang Li, Cong Shi, Yi Xie, Jian Liu, Bo Yuan, and Yingying Chen. Practical Adversarial Attacks Against Speaker Recognition Systems. In *Proceedings of ACM HotMobile*, pages 9–14, Austin, TX, USA, 2020.

[37] Zhuohang Li, Yi Wu, Jian Liu, Yingying Chen, and Bo Yuan. Advpulse: Universal, synchronization-free, and targeted audio adversarial attacks via subsecond perturbations. In *Proceedings of ACM SIGSAC CCS*, pages 1121–1134, Virtual Event, USA, 2020.

[38] Andy T Liu, Shang-Wen Li, and Hung-yi Lee. Tera: Self-supervised learning of transformer encoder representation for speech. *IEEE ACM Trans. Audio Speech Lang. Process.*, 29:2351–2366, 2021.

[39] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *Proceedings of ICLR*, Vancouver, BC, Canada, 2018.

[40] Alvin Martin and Mark Przybocki. The nist speaker recognition evaluation series. *National Institute of Standards and Technology Web site*, 2009.

[41] Mitchell McLaren, Victor Abrash, Martin Graciarena, Yun Lei, and Jan Pešán. Improving robustness to compressed speech in speaker recognition. In *Proceedings of ISCA INTERSPEECH*, pages 3698–3702, Lyon, France, 2013.

[42] Mitchell Mclaren, Diego Castán, Mahesh Kumar Nandwana, Luciana Ferrer, and Emre Yilmaz. How to train your speaker embeddings extractor. In *Proceedings of ISCA Odyssey*, pages 327–334, Les Sables d'Olonne, France, 2018.

[43] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, Ken Nakae, and Shin Ishii. Distributional smoothing with virtual adversarial training. *arXiv preprint arXiv:1507.00677*, 2015.

[44] Arsha Nagrani, Joon Son Chung, Weidi Xie, and Andrew Zisserman. Voxceleb: Large-scale speaker verification in the wild. *Computer Speech & Language*, 60:101027, 2020.

[45] D. O'Shaughnessy. Linear predictive coding. *IEEE Potentials*, (1):29–32, 1988.

[46] Monisankha Pal, Arindam Jati, Raghuveer Peri, Chin-Cheng Hsu, Wael AbdAlmageed, and Shrikanth Narayanan. Adversarial defense for deep speaker recognition using hybrid adversarial training. In *Proceedings of IEEE ICASSP*, pages 6164–6168, Toronto, ON, Canada, 2021.

[47] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. Specaugment: A simple data augmentation method for automatic speech recognition. *CoRR*, 2019.

[48] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukasˇ Burget, Ondˇrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlıˇcek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely. The Kaldi Speech Recognition Toolkit. In *Proceedings of IEEE ASRU*, Waikoloa, HI, USA, 2011.

[49] Mirco Ravanelli and Yoshua Bengio. Speaker recognition from raw waveform with sincnet. In *Proceedings of IEEE SLT Workshop*, pages 1021–1028, Athens, Greece, 2018.

[50] Ilyes Rebai, Yessine BenAyed, Walid Mahdi, and Jean-Pierre Lorré. Improving speech recognition using data augmentation and acoustic model fusion. *Procedia Computer Science*, 112:316–322, 2017.

[51] Research and Markets. Voice biometrics market forecast to 2028 - covid-19 impact and global analysis by component, type, authentication process, deployment, vertical, and application. https://www.researchandmarkets.com/reports/5623597/voice-biometrics-market-forecast-to-2028, 2021.

[52] Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-gan: Protecting classifiers against adversarial attacks using generative models. In *Proceedings of ICLR*, Vancouver, BC, Canada, 2018.

[53] Ali Shahin Shamsabadi, Francisco Sepúlveda Teixeira, Alberto Abad, Bhiksha Raj, Andrea Cavallaro, and Isabel Trancoso. Foolhd: Fooling speaker identification by highly imperceptible adversarial disturbances. In *Proceedings of IEEE ICASSP*, pages 6159–6163, Toronto, ON, Canada, 2021.

[54] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur. X-vectors: Robust dnn embeddings for speaker recognition. In *Proceedings of IEEE ICASSP*, pages 5329–5333, Calgary, AB, Canada, 2018.

[55] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *Proceedings of ICLR*, Banff, AB, Canada, 2014.

[56] Florian Tramèr, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. In *Proceedings of MIT Press NeurIPS*, pages 1633–1645, Virtual, 2020.

[57] Jesús Villalba, Yuekai Zhang, and Najim Dehak. x-Vectors Meet Adversarial Attacks: Benchmarking Adversarial Robustness in Speaker Verification. In *Proceedings of ISCA INTERSPEECH*, pages 4233–4237, Virtual Event, Shanghai, China, 2020.

[58] Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez Moreno. Generalized end-to-end loss for speaker verification. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4879–4883. IEEE, 2018.

[59] Qing Wang, Pengcheng Guo, Sining Sun, Lei Xie, and John HL Hansen. Adversarial Regularization for End-to-End Robust Speaker Verification. In *Proceedings of ISCA INTERSPEECH*, pages 4010–4014, Graz, Austria, 2019.

[60] Qing Wang, Pengcheng Guo, and Lei Xie. Inaudible Adversarial Perturbations for Targeted Attack in Speaker Recognition. In *Proceedings of ISCA INTERSPEECH*, pages 4228–4232, Virtual Event, Shanghai, China, 2020.

[61] Haibin Wu, Po-Chun Hsu, Ji Gao, Shanshan Zhang, Shen Huang, Jian Kang, Zhiyong Wu, Helen Meng, and Hung-Yi Lee. Adversarial Sample Detection for Speaker Verification by Neural Vocoders. In *Proceedings of IEEE ICASSP*, pages 236–240, Singapore, Singapore, 2022.

[62] Haibin Wu, Xu Li, Andy T. Liu, Zhiyong Wu, Helen Meng, and Hung-yi Lee. Adversarial defense for automatic speaker verification by cascaded self-supervised learning models. In *Proceedings of IEEE ICASSP*, pages 6718–6722, Toronto, ON, Canada, 2021.

[63] Haibin Wu, Yang Zhang, Zhiyong Wu, Dong Wang, and Hung-yi Lee. Voting for the right answer: Adversarial defense for speaker verification. In *Proceedings of ISCA INTERSPEECH*, pages 4294–4298, Brno, Czechia, 2021.

[64] Yi Xie, Cong Shi, Zhuohang Li, Jian Liu, Yingying Chen, and Bo Yuan. Real-time, universal, and robust adversarial attacks against speaker recognition systems. In *Proceedings of IEEE ICASSP*, pages 1738–1742, Barcelona, Spain, 2020.

[65] Weilin Xu, David Evans, and Yanjun Qi. Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks. In *Proceedings of ISOC NDSS*, San Diego, CA, 2018.

[66] Junichi Yamagishi, Christophe Veaux, and Kirsten MacDonald. CSTR VCTK Corpus: English Multi-speaker Corpus for CSTR Voice Cloning Toolkit, 2019. University of Edinburgh.

[67] Hitoshi Yamamoto, Kong Aik Lee, Koji Okabe, and Takafumi Koshinaka. Speaker Augmentation and Bandwidth Extension for Deep Speaker Embedding. In *Proceedings of ISCA INTERSPEECH*, pages 406–410, Graz, Austria, 2019.

[68] Ryuichi Yamamoto, Eunwoo Song, and Jae-Min Kim. Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In *Proceedings of IEEE ICASSP*, pages 6199–6203, Barcelona, Spain, 2020.

[69] Zhuolin Yang, Bo Li, Pin-Yu Chen, and Dawn Song. Characterizing Audio Adversarial Examples Using Temporal Dependency. In *Proceedings of ICLR*, Vancouver, BC, Canada, 2018.

[70] Tongqing Zhai, Yiming Li, Ziqi Zhang, Baoyuan Wu, Yong Jiang, and Shu-Tao Xia. Backdoor attack against speaker verification. In *Proceedings of IEEE ICASSP*, pages 2560–2564, Toronto, ON, Canada, 2021.

[71] Hanyi Zhang, Longbiao Wang, Yunchun Zhang, Meng Liu, Kong Aik Lee, and Jianguo Wei. Adversarial Separation Network for Speaker Recognition. In *Proceedings of ISCA INTERSPEECH*, pages 951–955, Virtual Event, Shanghai, China, 2020.

[72] Lei Zhang, Yan Meng, Jiahao Yu, Chong Xiang, Brandon Falk, and Haojin Zhu. Voiceprint mimicry attack towards speaker verification system in smart home. In *Proceedings of IEEE INFOCOM*, pages 377–386, 2020.

[73] Weiyi Zhang, Shuning Zhao, Le Liu, Jianmin Li, Xingliang Cheng, Thomas Fang Zheng, and Xiaolin Hu. Attack on practical speaker verification system using universal adversarial perturbations. In *Proceedings of IEEE ICASSP*, pages 2575–2579, Toronto, ON, Canada, 2021.

Algorithm 2 Adaptive Attack

```
Input: original audio $x$, target speaker embedding $e$
Output: adaptive audio AE $x'$
for $iterNum$ in $1$ to $MaxIter$ do
    $\nabla\delta \leftarrow BackProp(\mathcal{L}(SR(x+\delta),e),\delta)$
    $G \leftarrow \nabla\delta$
    for idx in 1 to 6 do
        if $d \in DirectSet$ then
            $\nabla\delta \leftarrow BackProp(\mathcal{L}(SR(d(x)+\delta),e),\delta)$
        else if $d \in BPDASet$ then
            $\nabla\delta \leftarrow BPDA(\mathcal{L}(SR(d(x)+\delta),e),\delta)$
        else if $d \in EOTSet$ then
            $\nabla\delta \leftarrow EOT(\mathcal{L}(SR(d(x)+\delta),e),\delta)$
        end if
        $G \leftarrow G + \nabla\delta$
    end for
    $\delta \leftarrow Clip(\delta - \alpha \cdot sign(\nabla\delta),\varepsilon)$
end for
$x' \leftarrow x + \delta$
return $x'$
```

A Adaptive Attack Algorithm

The algorithm of adaptive attack against *FraudWhistler* is detailed in Algorithm 2. In every iteration, we repeat the following steps. First, we obtain the gradient without distortion function by backpropagation as in a simple adversarial example attack, i.e., the defense is not considered. Second, for each distortion function employed in *FraudWhistler*, we use different gradient estimation methods to generate more gradients. Specifically, for codec, we directly use the backpropagation algorithm to obtain the gradient. For Quantization, we use BPDA to estimate the backpropagation gradient. For Noisifier, Reverber, DropChunk and DropFreq with randomness, we use EOT to estimate a robust backward gradient by internal multiple iterations of optimization. After that, all the obtained gradients are summed up and then clipped into the legitimate range of value.

Table 6: Adversarial attacks on SVs: "U/T/B" means untargeted attack, targeted attack, and both targeted and untargeted attacks. Con. Def. stands for considering detection mechanism. Optobj means optimization methods with a designed objective function. SV and SI stand for speaker verification and speaker identification.

	Attack Goal	Over-the-air	Universal	Con. Def.	Knowledge	Algorithms	Task
Kreuk *et al.* [30]	U	✗	✗	✗	white-box	FGSM	SV
UAPs [33]	B	✗	✓	✗	white-box	GenerativeNet	SI
Li *et al.* [35]	B	✗	✗	✗	white/black-box	FGSM	SV
Villalba *et al.* [57]	B	✗	✗	✗	white-box	FGSM/CW	SV
FoolHD [53]	B	✗	✗	✗	white-box	FGSM/BIM	SI
Li *et al.* [36]	B	✓	✗	✗	white-box	FGSM/Optobj	SI
AdvPulse [37]	T	✓	✓	✓	white-box	Optobj	SI
Xie *et al.* [64]	T	✓	✓	✗	white-box	Optobj	SI
FakeBob [7]	T	✓	✗	✓	black-box	NES+BIM	SI
Zhang *et al.* [73]	T	✓	✓	✗	white-box	PGD	SV
Wang *et al.* [60]	T	✗	✗	✗	white-box	Optobj	SI

Table 7: Defenses against AE attacks in the audio domain: A, P and D means adversarial training, purification, and detection respectively. NegEff stands for the negative effect on benign examples. Con. Adap. Atk means whether considering adaptive attacks. Res. means resist corresponding attacks.

	Retrain Model	NegEff	Plug-and-play	Con. Adap. Atk	Res. Universal	Res. Unseen	Type
REG [59]	✓	✓	✗	✗	✗	✓	A
HAT [46]	✓	✓	✗	✗	✗	✗	A
AS-Net [71]	✗	✓	✓	✗	✗	✗	P
Joshi *et al.* [25]	✗	✓	✓	✓	✗	✗	P
TERA [62]	✗	✓	✓	✗	✗	✗	P
Li *et al.* [34]	✗	✗	✗	✗	✓	✗	D
Voting [63]	✗	✗	✓	✗	✗	✗	D
Vocoder [61]	✗	✗	✓	✗	✗	✗	D

B Adversarial Example Attacks on Speaker Recognition

Existing AE attacks on SR systems are summarized in Table 6. We can observe that all of the works' attack goals include targeted attacks except one. Besides, less than half of the works consider the over-the-air scenario or universal attacks. Also, we can observe that only two of them consider defenses in their work, which implies the necessity of an effective adversarial example defense in the audio domain. Considering the knowledge of attackers, most of the listed works are under the white-box setting and most of the victim SR systems are speaker identification systems.

C Defenses Against Adversarial Example in Audio Domain

Existing defenses against AE attacks on the SR systems are summarized in Table 7. We can observe that some of the works need to retrain the SR model, inducing additional efforts. Besides, most of the works would introduce a negative effect on benign audio examples and some of the works could be plug-and-play as preprocessing modules. However, few works consider adaptive attacks as well as advanced attacks (e.g., universal attacks) and evaluate their defense on unseen adversarial example attacks, which is necessary for a practical defense to deploy in real-world scenarios.