Towards Understanding and Mitigating Audio Adversarial Examples for Speaker Recognition

Guangke Chen, Zhe Zhao, Fu Song, Sen Chen, Lingling Fan, Feng Wang, and Jiashui Wang

Abstract—Speaker recognition systems (SRSs) have recently been shown to be vulnerable to adversarial attacks, raising significant security concerns. In this work, we systematically investigate transformation and adversarial training based defenses for securing SRSs. According to the characteristic of SRSs, we present 22 diverse transformations and thoroughly evaluate them using 7 recent promising adversarial attacks (4 white-box and 3 black-box) on speaker recognition. With careful regard for best practices in defense evaluations, we analyze the strength of transformations to withstand adaptive attacks. We also evaluate and understand their effectiveness against adaptive attacks when combined with adversarial training. Our study provides thirteen useful insights and findings, many of them are new or inconsistent with the conclusions in the image and speech recognition domains, e.g., variable and constant bit rate speech compressions have different performance, and some non-differentiable transformations remain effective against current promising evasion techniques which often work well in the image domain. We demonstrate that the proposed novel feature-level transformation combined with adversarial training is rather effective compared to the sole adversarial training in a complete white-box setting, e.g., increasing the accuracy by 13.62% and attack cost by two orders of magnitude, while other transformations do not necessarily improve the overall defense capability. This work sheds further light on the research directions in this field. We also release our evaluation platform SPEAKERGUARD to foster further research.

Index Terms—Speaker recognition, adversarial defenses, adversarial examples, input transformation, adversarial training

1 INTRODUCTION

Speaker recognition (SR) is the process of automatically recognizing individual speakers by extracting and analyzing their unique acoustic characteristics. State-of-the-art speaker recognition systems (SRSs), based on machine learning (including deep learning), have been adopted by open-source platforms (e.g., Kaldi [1]) and commercial products (e.g., Microsoft Azure [2]), and used in safety-critical applications, e.g., remote voice authentication in financial transaction [3].

The popularity of SRSs has brought new security concerns. Recent studies have shown that both open-source and commercial SRSs are vulnerable to adversarial attacks [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], where the adversary adds an imperceptible noise to a voice from a source speaker such that the crafted adversarial voice is recognized as another speaker by the SRS. To thwart adversarial attacks, five input transformations [12], [13], [16], [17] (that transform inputs to disrupt adversarial perturbation before feeding them to models) and two adversarial training [6] (that augments the training data with adversarial examples to improve the robustness), derived from other domains, have been studied. However, these defenses are only evaluated against a few non-adaptive attacks. Thus, it is impossible to fairly compare their performance and also may lead to a false sense of robustness improvement [18], limiting their usage in practice. Indeed, these defenses become ineffective against adaptive attacks where the adversary is aware of the defenses and intends to circumvent them using evasion techniques from the image domain.

In this work, to secure SRSs against adversarial attacks, we systematically investigate transformation and adversarial training based defenses and thoroughly evaluate their effectiveness using both non-adaptive and adaptive attacks under the same settings.

To make the investigation comprehensive and systematic, and provide system maintainers more freedom and options to choose suitable defenses, we should cover as many diverse transformations as possible. To address this challenge, we study transformations according to the characteristics of audio signals and SRS's architecture. Different from images and image recognition systems, audio can be transformed at both waveform-level and feature-level, where at the waveform-level, audio can be transformed in the time- and frequency-domain while at the feature-level, different types of features in acoustic feature extraction pipeline can be transformed. To be diverse and comprehensive, we consider 22 diverse transformations (4 time-domain and 3 frequency-domain transformations, 7 audio compressions that transform audio at both time- and frequency-domains, and 8 novel feature compressions), covering all the 5 transformations studied in [12], [13], [16], [17]. Furthermore, from the respective of adaptive attacks for evasion, these transformations cover all the differentiable, non-differentiable, deterministic, and randomized types.

To thoroughly evaluate the defenses, we extend and implement all the recent promising adversarial attacks [4],

- *Guangke Chen is with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China, and with the SKLCS, Institute of Software, Chinese Academy of Sciences, Beijing, China.*
- *Zhe Zhao is with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China.*
- *Fu Song (corresponding author) is with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China. Email: songfu@shanghaitech.edu.cn*
- *Sen Chen is with the College of Intelligence and Computing, Tianjin University, Tianjin, China.*
- *Lingling Fan is with the College of Cyber Science, Nankai University, Tianjin, China.*
- *Feng Wang and Jiashui Wang are with the Ant Group, Zhejiang, China.*

[5], [6], [7], [12], [13], [14], [15], including 4 white-box attacks and 3 black-box attacks. The evaluation on 22 concrete attacks shows that the effectiveness of transformations does not necessarily decrease with increase of both distortion and attack strength, and their effectiveness varies with attacks, e.g., two time-domain transformations are more effective than others against L_∞ attacks (i.e., perturbations are limited in L_∞ norm) and feature-level transformations are often more effective than others against L_2 white-box attacks.

However, this evaluation does not provide security guarantees against a future adaptive adversary who has knowledge of defenses, so-called the adaptive attacks [18]. The challenge here is the design of the adaptive attacks. To avoid a possible false sense of robustness, adaptive attacks should be designed carefully to tailor to the specification of each defense [18]. We address this by taking into account the differentiability and randomness of transformations and utilizing Backward Pass Differentiable Approximation (BPDA) [19], Natural Evolution Strategy (NES) [20], and Expectation over Transformation (EOT) [21] to bypass non-differentiable and randomized transformations, respectively. We also design the Replicate adaptive attack targeting the compression operation of our proposed feature-level transformation. We remark that these evasion techniques have never been considered in the speaker recognition domain except that NES was adopted to estimate gradients by the black-box attack FAKEBOB [12]. The evaluation shows that (1) most transformations including the ones from [12], [13], [16], [17] become *ineffective*, (2) some non-differentiable audio compressions *cannot* be broken by BPDA which is promising in the image domain, (3) AAC and MP3 with *variable bit rate* are more difficult (resp. easier) to be bypassed than them with *constant bit rate* in the black-box (resp. white-box) setting; and (4) most of the *randomized* transformations remain resistant to black-box adaptive attacks.

To explore the effectiveness of transformations combined with adversarial training, we consider the promising adversarial training of [6] and evaluate the combined defenses under adaptive attacks. The evaluation shows that while the combination of a transformation and adversarial training does not necessarily bring the best of both worlds, the proposed feature-level transformation combined with adversarial training is very effective, improving the accuracy of both benign and adversarial examples in a complete white-box setting. We further evaluate this combined defense by varying various attack parameters. The results show that it is still effective, improving the accuracy by 13.62%, attack cost by two orders of magnitude, and distortion of adversarial examples, compared over vanilla adversarial training.

Throughout our study, another challenge is the lack of suitable and domain-specific platforms to enable large-scale, comprehensive, and rigorous evaluation. While there do exist platforms, e.g., Cleverhans [22] and ART [23], they focus on computer vision and cannot be well incorporated with SR models and datasets due to the special architecture (e.g., the acoustic feature extraction module) and the special pipeline (e.g., the enrollment phase) of SRSs. In addition, they do not provide any audio-specific defenses or imperceptibility metrics. To address this challenge, we built a platform SPEAKERGUARD.

In summary, we make the following main contributions.

- We perform the most comprehensive investigation of transformation based defenses for securing SRSs according to the characteristic of audio signals and SRS's architecture and study the impact of their hyper-parameters for mitigating adversarial voices without incurring too much negative impact on the benign voices.
- We thoroughly evaluate the proposed transformations for mitigating recent promising adversarial attacks on SRSs. With regard for best practices in defense evaluations, we carefully analyze their strength, on both models trained naturally and adversarially, to withstand adaptive attacks.
- Our study provides thirteen useful insights and findings, either newly reported or inconsistent with existing findings in other domains, which could advance research on adversarial examples in SR domain and assist the maintainers of SRSs to deploy suitable defense solutions to enhance their systems. Particularly, we find that our novel feature-level transformations combined with adversarial training is the most robust one against adaptive attacks.
- We develop the first platform SPEAKERGUARD for systematic and comprehensive evaluation of adversarial attacks and defenses on SRSs. It features mainstream SRSs, datasets, white- and black-box attacks, widely-used evasion techniques for adaptive attacks, evaluation metrics, and diverse defense solutions. We release our platform to foster further research in this direction (https://speakerguard.github.io).

2 BACKGROUND

Speaker Recognition Systems (SRSs). State-of-the-art SRSs use speaker embedding to represent acoustic characteristics of speakers as fixed-dimensional vectors. The typical speaker embedding is identity-vector (ivector) [24] based on the Gaussian Mixture Model (GMM) [25]. Recently, deep embedding was also proposed to compete with ivector. It uses deep learning to train a deep neural network from which speaker characteristics are extracted and represented as vectors, e.g. AudioNet [6], [26] and DeepSpeaker [27].

A generic architecture of SRSs is shown in Fig. 1, consisting of: training, enrollment, and recognition phases. In the training phase, a background model is trained using lots of voices from a large number of training speakers, representing the speaker-independent distribution of acoustic features. In the enrollment phase, the background model maps the voice uttered by each enrolling speaker to an *enrollment embedding*, regarded as the unique identity. In the recognition phase, given a voice of an unknown speaker, the *voice embedding* is extracted from the background model. The scoring module measures the similarity between the *enrollment embedding* and *voice embedding* based on which the decision module outputs the result. There are two typical scoring approaches: Probabilistic Linear Discriminant Analysis (PLDA) [28] and cosine similarity [29], where PLDA works well in most situations but needs to be trained using voices while cosine similarity is a reasonable substitution of PLDA without requiring training.

The acoustic feature extraction module converts the raw audio signals to acoustic features carrying characteristics of the raw audio signals. Common feature extrac-

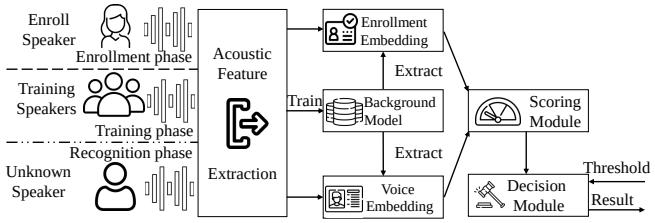


Fig. 1. Architecture of SRSs.

tion algorithms include Mel-Frequency Cepstral Coefficients (MFCC) [30] and Filter-Bank [30].

Recognition task. There are three main tasks: close-set identification (CSI), speaker verification (SV), and open-set identification (OSI). CSI identifies a speaker from a group of speakers. SV verifies if an input voice is uttered by the unique enrolled speaker, according to a preset threshold, where the input voice may be rejected by regarding the speaker as an imposter. OSI utilizes the scores and a preset threshold to identify which enrolled speaker utters the input voice, where if the highest score is less than the threshold, the input voice is rejected by regarding the speaker as an imposter. Moreover, CSI could be classified into two subtasks: CSI with enrollment (CSI-E) and CSI without enrollment (CSI-NE). CSI-E exactly follows the above description. In contrast, CSI-NE does not have the enrollment phase and the background model is directly utilized to identify speakers. Thus, ideally, a recognized speaker in CSI-NE task is involved in the training phase, while a recognized speaker in the CSI-E task should have enrolled in the enrollment phase but may not be involved in the training phase.

Threat model. An adversarial attack on an SRS aims to craft an adversarial voice by adding an imperceptible perturbation to a given voice uttered by a source speaker, so that the SRS under attack misclassifies it as another speaker. According to the adversary's knowledge about the SRS under attack, we classify attacks into *white-box* and *black-box* attacks. The adversary for a white-box attack has full access to SRS architecture, parameters, etc., while the adversary for a black-box attack does not have any information about the SRS but can access the target model as an oracle, i.e., providing a series of carefully crafted inputs to the SRS and observing its outputs. According to the adversary's knowledge about the deployed defenses, we classify attacks into *non-adaptive* and *adaptive* attacks. The adversary for the non-adaptive attack is unaware of the deployed defense, so he crafts adversarial voices without consideration for the defense, while the adversary for the adaptive attack has complete knowledge of the defense (e.g., its implementation detail and concrete values for any tunable parameter) and intends to bypass it. Different combinations of knowledge about the SRS and deployed defense leads to four attack scenarios considered in this work, namely, *white-box non-adaptive*, *black-box non-adaptive*, *white-box adaptive*, and *black-box adaptive attacks*.

3 Defenses

3.1 Motivation

Recently, adversarial attacks on speaker recognition have been extensively studied [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14]. Results show that both state-of-the-art open-source and commercial SRSs can be fooled by adding small perturbations to the original voice, even playing over the air in the physical world.

In the image and speech recognition domains, studies have proposed transformation based defenses that apply certain transformations to inputs before feeding them to the model for recognition in order to recover benign counterparts from adversarial examples, e.g., [31], [32]. While such defenses are effective for defending against non-adaptive attacks, they may be evaded by adaptive attacks [18]. Nevertheless, some transformations (but not all) achieve promising results when combined with adversarial training even in a complete white-box setting [18], [33]. However, the same conclusion cannot be drawn on speaker recognition without a careful and rigorous evaluation, because of the difference between speaker recognition and image/speech recognitions. Compared with image recognition systems, SRSs have complicated architectures and individual components, in particular, the acoustic feature extraction pipeline. Also, while the well-trained vision model is directly exploited to classify input images into one of the training classes, the well-trained background model of SRSs is adapted to speaker-specific models during enrollment and used to map input utterances into identity embeddings during recognition, since the enrolled and inference speakers are not necessarily involved in the training phase. While speech recognition minimizes speaker-dependent variations to determine the underlying text or command, speaker recognition treats the phonetic variations as extraneous noise to determine the source of the speech signal. All these differences may lead to inconsistent conclusions in the speaker recognition domain with other domains. In fact, we indeed found such inconsistent findings (cf. Section 7).

Therefore, in the speaker recognition domain, five input transformation [12], [13], [16], [17] and two adversarial training [6] based defenses have been studied. Though promising, these defenses are only evaluated against *few* attacks on different models, recognition tasks, and datasets, let alone adaptive attacks [18] and combinations of transformation and adversarial training. Thus, it is impossible to fairly compare their performance and also may lead to a false sense of robustness improvement brought by defenses without considering adaptive attacks, limiting their usage in practice. It is also unclear if combining a transformation with adversarial training results in a more effective defense, as many existing defenses combined with adversarial training result in lower robustness than adversarial training on its own in the image domain [18]. Therefore, *there is a lack of comprehensive investigation and rigorous quantitative understanding of defenses on speaker recognition, in particular, effective defenses*. This work is aimed at filling this gap.

3.2 Design Overview

According to the architecture of SRSs (cf. Fig. 1), we should consider both robust training and input transformation, where the former is conducted during the training phase and the latter takes effect in the recognition phase. When combined, they may lead to a more robust defense. For input transformation, we design audio transformations based on

the following two key characteristics of speaker recognition, compared over image recognition.

Architecture characteristic. For state-of-the-art neural network based image recognition, an image is directly fed to a system without feature engineering. Due to the time-varying non-stationary property of voices, voices are not resilient enough to noises and other variations, and audio waveform signals themselves cannot effectively represent speaker characteristics [34]. Hence, to achieve better feature representative capacity and system performance [35], a modern SRS has an acoustic feature extraction pipeline for extracting acoustic feature from waveforms (cf. Fig. 1). This gives rise to waveform-level input transformations (W-transformations) and feature-level input transformations (F-transformations).

Audio signal characteristic. While images are naturally two-dimensional, raw audio samples form a one-dimensional time series signal [36]. Even though audio signals are often transformed into two-dimensional time-frequency representations, the two axes, time and frequency, fundamentally differ from the horizontal and vertical axes in an image. Furthermore, images are commonly analyzed as a whole or in patches with little order constraints while audio signals have to be analyzed sequentially in chronological order. These properties give rise to audio-specific W-transformations that can be performed either in time-domain or frequency-domain.

Based on the above characteristics, to be diverse and comprehensive, we investigate both W-transformations and F-transformations, while for the former, we consider both time-domain and frequency-domain ones. When necessary and possible, we also evaluate the effectiveness of transformations combined with robust training. When devising an input transformation based defense, it is also important to consider if it is differentiable[1] and deterministic, due to the fact that most white-box attacks leverage gradient to craft adversarial examples. In general, non-differentiable input transformations are more difficult to evade than differentiable ones, and randomized input transformations are more difficult to evade than deterministic ones. Thus, all the types should be addressed to understand their effectiveness. All the transformations we considered are summarized in Table 1, covering differentiable, non-differentiable, deterministic, and randomized types.

TABLE 1: Transformations

		Name	Parameters	D	R
Waveform Level	Time Domain	**Quantization (QT)** [31]	q: quantized factor	✗	✗
		Audio Turbulence (AT) [40]	SNR: signal-to-noise ratio	✓	✓
		Average Smoothing (AS) [13]	k: kernel size	✓	✗
		Median Smoothing (MS) [31]	k: kernel size	✓	✗
	Frequency Domain	**Down Sampling (DS)** [31]	τ: downsampling freq.	✓	✗
		Low Pass Filter (LPF) [41]	f_p: passband edge freq. f_s: stopband edge freq.	✓	✗
		Band Pass Filter (BPF) [42]	f_{pl}, f_{pu}: passband edge freq. f_{sl}, f_{su}: stopband edge freq.	✓	✗
	Speech Compression	**OPUS**	b_o: compression bitrate	✗	✗
		SPEEX	b_s: compression bitrate	✗	✗
		AMR	b_r: compression bitrate	✗	✗
		AAC-V	q_c: quality	✗	✗
		AAC-C	b_c: compression bitrate	✗	✗
		MP3-V	q_m: quality	✗	✗
		MP3-C	b_m: compression bitrate	✗	✗
Feature Level		**FEATURE COMPRESSION (FeCo)** 4 feature types × 2 compression alg.	cl_m: cluster method cl_r: cluster ratio	✓	✓

Note: D=Differentiable and R=Randomized.

3.3 Robust Training

Robust training strengthens the resistance of a model to adversarial examples during training. We adopt adversarial training, one of the most effective techniques in the image domain, which augments the training data with adversarial examples. Formally, adversarial training intends to find the model parameter θ which minimizes the following loss:

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}[\texttt{max}_{\delta\in S} f(\theta, x+\delta, y)] \approx \frac{1}{n}\sum_{i=1}^{n} \texttt{max}_{\delta\in S} f(\theta, x_i+\delta, y_i)$$

where S is the set of allowed perturbations, $\mathcal{D}$ is the underlying data distribution over pairs of samples x and corresponding labels y, $\{(x_i, y_i)\}_{i=1}^n$ is the training dataset that mimics the data distribution $\mathcal{D}$, and f is the training loss function, typically the cross-entropy loss. Efficient adversarial attacks such as FGSM [38] and PGD [39] are widely used to solve the above maximization problem.

3.4 W-Transformations

For W-transformations, we consider both time-domain and frequency-domain ones. We also consider various speech compression which can be seen as W-transformations performed both in the time- and frequency-domains.

Time-domain W-transformations. We study four time-domain W-transformations, inspired by image input transformations [31]. (1) Quantization (QT) rounds the amplitude of each sample point of a voice to the nearest integer multiple of a factor q, intended to disrupt the adversarial perturbation since its amplitude is usually small in the input space. (2) Audio turbulence (AT) adds random noise to an input voice in an element-wise way to disrupt the adversarial perturbation which is assumed to be sensitive to noise. The magnitude of the noise is adjusted by signal-to-noise ratio (SNR) $10\log_{10}\frac{P_I}{P_n}$ where P_I (resp. P_n) is the power of input voice (resp. random noise). (3) Average smoothing (AS) and (4) median smoothing (MS) mitigate adversarial examples by smoothing the waveform of the input voice. A mean (resp. median) smooth with kernel size k (must be odd) replaces each element x_k with the *mean* (resp. *median*) value of its k neighbors. We remark that QT is non-differentiable due to the round operation while the others are differentiable, and AT is randomized while the others are deterministic.

Frequency-domain W-transformations. We consider three W-transformations in frequency-domain, all of which are differentiable and deterministic. (1) Down sampling (DS) down-samples voices and applies signal recovery to disrupt adversarial perturbations. The down-sample frequency is determined by the ratio, denoted by τ, between the new and original sampling frequencies. (2) Low pass filter (LPF) assumes that human voices are within relatively lower frequencies than adversarial perturbation, and applies a low-pass filter to remove the high-frequent perturbations. A low-pass filter has two parameters: the edge frequencies

1. Differentiable here means that a transformation can be implemented in frameworks (e.g., Pytorch) that supports auto-differentiation [37], i.e., enabling back-propagation of gradients and providing informative gradients for adversarial example generation. Though it is non-rigorous, we use it to keep consistent with [18].

of the passband (f_p) and the stopband (f_s). (3) Band pass filter (BPF) combines LPF with a high-pass filter to remove both high-frequent and low-frequent perturbations. BPF has four parameters: the lower and upper edge frequencies of the passband (f_{pl} and f_{pu}), the lower and upper cutoff frequencies of the stopband (f_{sl} and f_{su}). We remark that these transformations are derived from the speech recognition domain [31], [41], [42], but only DS has been applied in the speaker recognition against two black-box attacks FAKEBOB [12] and SirenAttack [13].

Speech compression. Based on the psychoacoustic principle, speech compression aims to suppress redundant information within a speech to improve storage or transmission efficiency. When an adversarial perturbation is redundant, it can be eliminated by speech compression. Speech compression achieves the aforementioned purpose by reducing the bit rate, thus can be seen as transformations performed both in the time- and frequency-domains. We investigate 7 standard lossy speech compression techniques, grouped into two categories: Constant Bit Rate (CBR) and Variable Bit Rate (VBR). The former uses a fixed bit rate and the latter exploits a dynamic bit rate schedule controlled by the quality parameter. We consider OPUS [43], SPEEX [44], AMR [45], AAC-C [46], and MP3-C [47] for CBR, and AAC-V [46] and MP3-V [47] for VBR. These transformations are non-differentiable and deterministic.

3.5 F-Transformations

The design of F-transformations is motivated by the following research questions: (Q1) *What kind of acoustic features can be transformed?* and (Q2) *How to transform them?*.

To address Q1, we have to understand what kind of features are used in SRSs. Fig. 2 shows a typical flow of feature processing. First, the *original features* (e.g., MFCC or FilterBank) are extracted from an input raw waveform. Next, to capture temporal information, time-derivative features [35] are successively extracted from and added into the original features, leading to the *delta features*. After that, cepstral mean and variance normalization (CMVN) [48] is applied to reduce channel and reverberation effects, resulting in *cmvn features*. Finally, voice activity detection (VAD) [49] is utilized to remove the unvoiced frames, resulting in *final features*. Therefore, four types of features could be transformed.

To address Q2, a straightforward idea is to extend W-transformations. However, (1) W-transformations work on audio waveforms in two-dimensional time-frequency representations, while acoustic features are represented by a matrix, one row of features per frame. It prevents frequency-domain W-transformations and speech compression from being extended. (2) The mapping from waveforms to features is not linear, and a small perturbation in the input voice may lead to a large perturbation at the feature level. This difference refuses time-domain W-transformations where adversarial perturbations are assumed to be small and/or sensitive to noise.

We propose FEATURE COMPRESSION (FeCo) to disrupt adversarial perturbations at the feature level. We regard each feature matrix $\mathcal{M}$ with N frames and each frame $\mathbf{a}_i$ consisting of d features as N data points in d-dimensional space and compute a compressed feature matrix with K

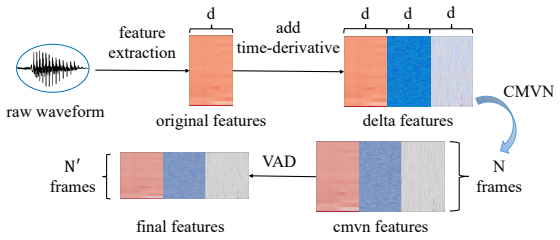


Fig. 2. A typical flow of feature processing.

Algorithm 1 FeCo

Input: feature matrix $\mathcal{M} = [\mathbf{a}_1, \cdots, \mathbf{a}_N]$; cluster ratio $0 < cl_r < 1$; cluster oracle $\mathcal{O}$ = kmeans or warped-kmeans
Output: compressed feature matrix $\mathcal{M}'$

1: $K \leftarrow \lceil N \times cl_r \rceil$ ▷ K = number of clusters
2: $[b_1, \cdots, b_N] \leftarrow \mathcal{O}(\mathcal{M}, K)$ ▷ $\mathbf{a}_i$ is assigned to the b_i-th cluster
3: **for** $(i = 1; i \leq K; i++)$ **do**
4: $C_i \leftarrow \{\mathbf{a}_k \mid b_k = i\}$ ▷ compute the i-th cluster
5: $\mathbf{m}_i \leftarrow \frac{1}{|C_i|} \sum_{\mathbf{a} \in C_i} \mathbf{a}$ ▷ compute the representative vector
6: $\mathcal{M}' \leftarrow [\mathbf{m}_1, \cdots, \mathbf{m}_K]$ ▷ concatenate the representative vectors
7: **return** $\mathcal{M}'$

frames for $K < N$. Our idea is described in Algorithm 1. The number K of clusters is first computed according to the given cluster ratio cl_r (line 1). Then, we partition N frames into K clusters by invoking the cluster oracle $\mathcal{O}$ (line 2), which returns a list of indices $b_1, \cdots, b_N$ such that each frame $\mathbf{a}_i$ is assigned to the b_i-th cluster. Next, each cluster C_i is represented by a representative vector $\mathbf{m}_i$ (line 5). Finally, K representative vectors are combined to form the new feature matrix $\mathcal{M}'$.

To partition N frames into K clusters, various clustering methods, e.g., kmeans [50] and fuzzy-kmeans [50], could be leveraged. In this work, we use kmeans and its variant warped-kmeans [51] and leave others as future work. Compared to kmeans, warped-kmeans preserves the temporal dependency of the data by imposing some constraints on the partition operation, thus is more suitable to cluster sequential data. Both kmeans and warped-kmeans use the average of all the frames in one cluster as the representative.

Algorithm 1 could be applied to any of original, delta, cmvn, and final features. We use FeCo-o, FeCo-d, FeCo-c, and FeCo-f to denote these four concrete F-transformations, all of which are randomized and differentiable. The randomness of FeCo lies in the initialization of kmeans and warped-kmeans algorithms. At the beginning, they *randomly* select K vectors from N vectors as the initial cluster centers, which will be used in the later clustering operations. Different initialization may produce different clustering results (Line 2), thus leading to different feature matrix $\mathcal{M}'$.

4 EVALUATION SETUP AND METRICS

4.1 Main Evaluation Setup

To evaluate defenses against adversarial voices on SRSs, we developed a platform, named SPEAKERGUARD.

Models. We use two mainstream SRSs: a pre-trained model ivector-PLDA [52] from the popular open-source platform KALDI having 11.5k stars and 4.9k forks on GitHub [1] and a one-dimension convolution neural network based model AudioNet [26]. Both ivector-PLDA and AudioNet were used as the SRS under attack in many prior works,

TABLE 2: SR models

	ivector-PLDA [52]	**AudioNet** [26]
Embedding & Feature types	T & MFCC	D & Filter-Bank
Add 1st & 2nd time-derivative	✓	✗
Apply CMVN & VAD	✓	✗
#Feature dim	72	32
Training algorithm	US	S
Scoring method	PLDA	-

Note: T/D means GMM/deep model and (U)S means (un)supervised learning.

TABLE 3: Voice datasets

	Spk$_{10}$-enroll	**Spk$_{10}$-test**	**Spk$_{251}$-train**	**Spk$_{251}$-test**
Task	CSI-E/SV/OSI		CSI-NE	
#Speakers	10 (5M,5F)	10 (5M,5F)	251 (126M,125F)	251 (126M,125F)
#Voices	10×10	100×10	25652	2887
Length	3–21s (7.2s)	1–15s (4.3s)	1–24s (12.3s)	1–19s (11.7s)

Note: x-y (z) denotes that the minimal, maximal and average length of voices, and nM/mF denotes that the number of male/female speakers is n/m.

e.g., [5], [12], [53], [54], [55] for ivector-PLDA and [6], [53], [56] for AudioNet. Both of them have excellent performance on benign voices (cf. Baselines in Tables 4 and 7). Details of two models are shown in TABLE 2. Due to the massive experiments, we only target the CSI task (i.e., CSI-E and CSI-NE). The results on the SV and OSI tasks could be similar, as demonstrated in [12].

Datasets. We use four datasets derived from Librispeech [57]: Spk$_{10}$-enroll, Spk$_{10}$-test, Spk$_{251}$-train, and Spk$_{251}$-test. The datasets are summarized in TABLE 3 (details refer to Supplemental Material A.1.)

Attacks. To thoroughly evaluate the defenses, we implement 4 promising white-box attacks (i.e., FGSM [38], PGD [39], CW$_\infty$, and CW$_2$ [58]), and 3 state-of-the-art black-box attacks (i.e., FAKEBOB [12], SirenAttack [13], and Kenansville [15]). All of them craft adversarial voices via solving optimization problems using L_∞ norm to limit perturbations, except that Kenansville is a signal processing-based decision-only attack and CW$_2$ minimizes adversarial perturbations in the loss function using L_2 norm. To solve the optimization problems, FGSM, PGD, CW$_\infty$, and CW$_2$ use gradients, FAKEBOB uses gradient-estimation, and SirenAttack uses the gradient-free particle swarm optimization. Note that CW$_\infty$ uses the loss function of the CW attack but optimized by PGD, the same as [39], to improve the attack efficiency. Details refer to Supplemental Material A.2.

To avoid fake adversarial voices due to the discretization problem [59], i.e., adversarial voices become benign after being transformed into concrete voices, they are evaluated after storing back into the 16-bit PCM form. We only consider untargeted attacks which are more challenging to be defeated than targeted attacks [19]. Since SRSs only take waveforms as input in practice, we implement all the attacks to add perturbations directly to the waveforms rather than the acoustic features as in [5] where the adversarial acoustic features must be reconstructed back to waveforms, which is a lossy procedure, thus weakening the attack's effectiveness and imperceptibility [60].

We use a machine with Ubuntu 18.04, an Intel Xeon E5-2697 v2 2.70GHz CPU, 376GiB memory, and a GeForce RTX 2080Ti GPU.

4.2 Evaluation Metrics

Attack effectiveness. To evaluate the effectiveness of an attack, we use model accuracy on adversarial examples (A_a), i.e., the proportion of adversarial examples that are correctly classified by the model. Thus, smaller A_a indicates better attack. Note that $100\% - A_a$ is the untargeted attack success rate.

Defense effectiveness. A usable defense should not only improves resistance to adversarial examples, but also sacrifices accuracy on benign examples as little as possible. Thus, we measure the effectiveness of a defense using model accuracy on adversarial examples (A_a) and model accuracy on benign examples (A_b), respectively, where the larger A_a (resp. A_b) is, the better the defense is. We also use the R1 score, $R1 = \frac{2\times A_b \times A_a}{A_b + A_a}$ [42], which assigns equal importance to A_b and A_a, to quantify the usability of a defense.

Imperceptibility. To measure the imperceptibility, we use Signal-to-Noise Ratio (SNR) [40] and Perceptual Evaluation of Speech Quality (PESQ) [61]. SNR is defined as $10\log_{10}\frac{P_x}{P_\delta}$, where P_x (resp. P_δ) is the power of the original voice (resp. perturbation). PESQ is one of the objective perceptual measures, simulating human auditory system [62]. The calculation of PESQ is more involved. It first applies an auditory transform to obtain the loudness spectra of the original and adversarial voices, and then compares two loudness spectra to obtain a metric score whose value is in the range of -0.5 to 4.5. We refer readers to [61] for more details. Larger SNR and higher PESQ indicate better imperceptibility.

5 Evaluation of Transformations

5.1 Evaluation Setup

We limit the perturbation budget ϵ to 0.002 for L_∞ attacks, the same as [6], [12], unless explicitly stated. The number of steps for PGD and CW$_\infty$ range from 10 to 50 with step_size $\alpha = \frac{\epsilon}{5} = 0.0004$ for each step. For CW$_2$, we set the initial trade-off constant c to 0.001, use 9 binary search steps to minimize perturbations, run 900-9000 iterations to converge, and vary the confidence parameter κ from 0, 2, 5, 10, 20 to 50. For FAKEBOB, we limit the number of iterations to 200 with the parameter samples_per_draw of NES $m = 50$ and $\kappa = 0.5$. For SirenAttack, we use the optimal parameters reported in [13], i.e., the maximum number of epochs $epoch_{max} = 300$, the iteration limit of the PSO subroutine $iter_{max} = 30$, and the number of particles $n\_particles = 25$. For Kenansville, we use the SSA method to perturb a voice and set the maximal attack factor max_attack_factor to 100 and maximal number of iterations $max\_iteration$ to 30, which is sufficient for the attack to converge according to our experiments. FFT method is not considered since it is much less effective than the SSA method [63].

We consider the ivector-PLDA model for the CSI-E task which is enrolled with 10 speakers using the Spk$_{10}$-enroll dataset. We use the Spk$_{10}$-test dataset to test the model, resulting in 99.8% accuracy on benign examples. We also use the Spk$_{10}$-test dataset to craft adversarial examples. Though the ivector-PLDA model is pre-trained without any transformations, it still produces sufficient accuracy on benign examples, as shown in column (A_b) of TABLE 4. Thus, we do not re-train it when transformations are deployed. As each transformation contains at least one tunable parameter which may affect the effectiveness, we tune parameters and choose the best ones according to their R1 scores for the remaining experiments. Details are given in Supplemental Material A.3.

TABLE 4: Results of transformations against non-adaptive attacks

Defense	R1 Score	A_b	A_a																				
			L_∞ white-box attacks													L_2 white-box attacks						black-box attacks		
			FGSM	PGD						CW_∞						CW_2						Score-based (L_∞)		Decision-only
				10	20	30	40	50	100	10	20	30	40	50	100	0	2	5	10	20	50	FAKEBOB	SirenAttack	Kenansville
Baseline	8.3%	99.8%	42.3%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	6.5%	0%	0%	0%	0%	0%	19.8%	18.7%	8.0%
QT	**76.0%**	**86.8%**	**76.8%**	**61.2%**	**55.4%**	**56.6%**	**62.5%**	**59.8%**	**67.2%**	**60.7%**	**55.0%**	**55.8%**	**62.2%**	**57.4%**	**65.0%**	86.8%	86.1%	86.4%	86.2%	**84.9%**	**49.9%**	**91.3%**	88.2%	31.7%
AT	**84.5%**	89.2%	**82.9%**	**77.8%**	**75.9%**	**75.6%**	**78.5%**	**76.6%**	**81.2%**	**77.9%**	**76.7%**	**74.2%**	**77.8%**	**75.5%**	**81.2%**	89.1%	89.0%	89.1%	**89.2%**	**88.9%**	**78.5%**	**95.4%**	94.0%	**40.6%**
AS	39.8%	98.1%	**46.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**96.8%**	**95.0%**	87.4%	65.5%	**20.1%**	**0.0%**	**47.5%**	**69.3%**	21.9%
MS	53.9%	**83.9%**	65.6%	**21.3%**	**17.1%**	**17.3%**	**22.1%**	**18.3%**	**24.5%**	**21.2%**	**17.9%**	**17.1%**	**23.6%**	**18.9%**	**24.4%**	**77.1%**	76.4%	73.2%	68.8%	57.9%	26.9%	71.5%	**70.6%**	**41.8%**
DS	38.3%	91.8%	57.2%	0.3%	0.2%	0.2%	0.2%	0.1%	0.2%	0.2%	0.3%	0.3%	0.1%	0.2%	0.3%	**77.2%**	**73.4%**	**68.1%**	59.9%	39.3%	0.7%	67.3%	**66.8%**	20.2%
LPF	38.2%	96.9%	59.8%	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	84.6%	78.2%	71.7%	59.7%	22.2%	**0.0%**	54.3%	81.5%	**10.6%**
BPF	**36.1%**	91.0%	51.4%	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	0.1%	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**79.0%**	75.9%	68.5%	**52.9%**	21.2%	0.2%	58.5%	76.8%	11.6%
OPUS	56.8%	**88.6%**	67.9%	17.4%	14.1%	15.0%	17.9%	17.1%	23.3%	17.0%	14.2%	15.3%	18.5%	17.9%	22.4%	84.0%	82.9%	81.0%	78.8%	71.8%	31.5%	87.5%	86.0%	**37.2%**
SPEEX	53.5%	93.8%	**71.8%**	7.2%	6.6%	7.9%	11.9%	10.6%	21.8%	6.7%	6.8%	7.8%	11.3%	9.6%	22.5%	88.1%	87.5%	84.0%	77.4%	59.6%	18.3%	87.9%	89.0%	30.0%
AMR	55.4%	96.8%	67.4%	6.4%	7.0%	7.7%	11.0%	8.1%	15.9%	5.8%	8.0%	7.8%	11.4%	8.7%	17.3%	94.8%	93.7%	**92.3%**	**88.6%**	67.2%	24.6%	**94.2%**	93.6%	22.9%
AAC-V	**29.8%**	**99.8%**	**47.1%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	89.7%	**72.4%**	**37.3%**	**5.9%**	**0.0%**	**0.0%**	**34.6%**	87.5%	**10.4%**
AAC-C	44.7%	92.7%	64.2%	2.8%	2.3%	1.8%	2.5%	2.4%	2.7%	3.2%	2.3%	1.6%	2.6%	2.2%	2.6%	83.6%	82.3%	78.5%	71.8%	51.1%	8.1%	76.4%	89.8%	12.6%
MP3-V	**27.1%**	**99.6%**	**48.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	**0.0%**	87.4%	**62.2%**	**15.9%**	**0.3%**	**0.0%**	**0.0%**	**32.0%**	90.5%	**8.9%**
MP3-C	40.9%	96.4%	53.1%	**0.0%**	**0.0%**	0.1%	0.1%	**0.0%**	**0.0%**	**0.0%**	**0.0%**	0.1%	0.1%	**0.0%**	**0.0%**	87.6%	84.3%	79.0%	63.9%	29.3%	0.4%	71.1%	91.1%	11.4%
FeCo-o(k)	**58.0%**	94.0%	70.4%	16.3%	13.8%	13.0%	17.0%	12.7%	20.8%	14.1%	14.2%	15.2%	15.1%	10.7%	22.5%	91.4%	86.1%	86.5%	83.4%	**74.0%**	**42.0%**	85.5%	92.1%	26.7%
FeCo-d(k)	49.6%	**99.4%**	70.5%	0.2%	**0.0%**	0.2%	0.9%	0.3%	1.1%	0.1%	0.1%	0.5%	0.6%	0.8%	1.0%	**97.1%**	**94.4%**	**94.1%**	**87.3%**	62.8%	14.7%	85.6%	**94.4%**	20.1%
FeCo-c(k)	48.2%	98.8%	68.8%	**0.0%**	0.2%	0.1%	0.1%	0.1%	0.5%	0.1%	0.3%	0.1%	0.2%	0.3%	1.0%	**96.3%**	**93.8%**	**91.0%**	82.1%	55.0%	11.2%	84.0%	**95.1%**	20.8%
FeCo-f(k)	47.2%	98.2%	67.1%	0.3%	0.3%	0.5%	0.3%	0.4%	0.9%	0.5%	0.5%	0.6%	0.6%	0.7%	1.0%	93.4%	90.3%	86.6%	78.7%	51.2%	10.8%	83.6%	**94.8%**	21.0%
FeCo-o(wk)	50.5%	96.7%	66.6%	3.9%	3.5%	3.7%	4.3%	4.0%	6.5%	3.6%	3.2%	4.4%	4.8%	3.9%	7.0%	91.3%	88.5%	84.4%	77.5%	58.5%	26.8%	89.6%	91.7%	24.0%
FeCo-d(wk)	49.8%	98.2%	70.2%	1.7%	1.1%	1.1%	3.0%	1.8%	3.5%	1.4%	0.6%	1.1%	2.7%	2.0%	2.7%	93.9%	90.9%	88.3%	82.9%	64.0%	23.4%	88.1%	88.7%	19.9%
FeCo-c(wk)	48.5%	98.0%	68.3%	1.4%	0.7%	0.7%	2.4%	1.3%	2.5%	1.1%	0.6%	1.0%	2.3%	1.8%	1.8%	93.0%	89.0%	87.1%	79.4%	58.6%	20.1%	87.6%	87.9%	21.2%
FeCo-f(wk)	49.0%	97.6%	68.5%	2.0%	1.2%	0.8%	3.0%	1.5%	3.0%	2.2%	1.5%	1.2%	2.0%	2.2%	3.2%	91.6%	88.7%	85.7%	79.5%	60.4%	22.1%	88.7%	88.8%	22.1%

Note: k (resp. wk) denotes kmeans (resp. warped-kmeans). The top-3 highest/lowest results are highlighted in blue/red color except for Baseline where no defense is deployed. The accuracy A_a used for computing R1 Score is the average of all the attacks in the same row.

TABLE 5: Imperceptibility and strength of non-adaptive attacks

Attack		Imperceptibility		Loss	
		SNR	PESQ	$\mathcal{L}_{\mathbf{CE}}$	$\mathcal{L}_{\mathbf{M}}$
FGSM		28.53	2.23	3.91	-1.66
PGD-x	**x=10**	32.77	2.85	45.88	-45.87
	x=20	31.57	2.72	54.50	-54.50
	x=30	31.42	2.70	58.38	-58.38
	x=40	31.45	2.71	60.52	-60.52
	x=50	31.31	2.69	62.23	-62.23
	x=100	31.29	2.70	67.10	-67.10
CW∞-x	**x=10**	32.74	2.85	44.59	-44.56
	x=20	31.88	2.76	53.21	-53.19
	x=30	31.62	2.73	57.36	-57.35
	x=40	31.51	2.72	59.94	-59.93
	x=50	31.45	2.71	61.04	-61.03
	x=100	31.38	2.71	66.36	-66.36
CW_2-κ	**κ=0**	52.99	4.24	1.54	-1.12
	κ=2	51.42	4.19	2.94	-2.87
	κ=5	49.73	4.10	6.42	-6.35
	κ=10	47.09	3.95	11.28	-11.31
	κ=20	42.14	3.60	21.70	-21.25
	κ=50	30.44	2.46	51.88	-51.43
FAKEBOB		31.40	2.71	0.91	-0.10
SirenAttack		31.03	2.66	0.91	-0.10
Kenansville		8.73	1.87	3.32	-2.82

Note: $\mathcal{L}_{\mathbf{CE}}$ and $\mathcal{L}_{\mathbf{M}}$ respectively denote cross entropy loss and margin loss. The larger $\mathcal{L}_{\mathbf{CE}}$ (resp. the smaller $\mathcal{L}_{\mathbf{M}}$), the stronger the attack.

5.2 Results

The results are reported in TABLE 4, where row (Baseline) shows the accuracy without any defense, indicating the effectiveness of attacks. In general, the effectiveness of transformations significantly varies with attacks. The results provide many interesting and useful findings, including but not limited to the following ones.

Effectiveness versus level/domain. Time-domain W-transformations (e.g., QT, AT and MS) are often more effective than others on L_∞ attacks, while F-transformations are often more effective than others on L_2 attacks. Among W-transformations, FeCo-o and FeCo-d often perform slightly better than others, as transformation on preceding features also affects succeeding features. Between kmeans and warped-kmeans, the effectiveness varies with attacks and in general they are almost comparable. In terms of R1 score, FeCo-o with kmeans, i.e., FeCo-o(k), ranks the first place.

Findings 1. Time-domain (resp. feature-level) transformations are often more effective than others on L_∞ (resp. L_2) attacks.

Effectiveness versus distortion. Almost all the transformations perform better against FGSM, FAKEBOB, Kenansville and SirenAttack attacks than PGD, CW_∞, and CW_2-50 attacks. To find out the reason for this difference, we report the imperceptibility and strength of non-adaptive attacks in TABLE 5. According to the imperceptibility metrics SNR and PESQ, we observe that FGSM, SirenAttack, and Kenansville (resp. FAKEBOB) attacks introduce larger (resp. comparable) levels of distortion than PGD, CW_∞, and CW_2 attacks. This indicates that there is no direct correlation between the distortion of adversarial voices and the effectiveness of input transformations. In contrast, according to the loss values of $\mathcal{L}_{\mathbf{CE}}$ and $\mathcal{L}_{\mathbf{M}}$, we observe that the single-step attack FGSM and the black-box attacks (i.e., FAKEBOB, SirenAttack, and Kenansville) are much weaker than PGD, CW_∞, and CW_2 attacks. In fact, FGSM is a single-step attack, FAKEBOB and SirenAttack adopt an early-stop strategy, and Kenansville is a decision-based attack, so adversarial examples crafted by them are weak (i.e., close to the decision boundary), while PGD, CW_∞, and CW_2-50 continue searching for strong adversarial examples (i.e., far from the decision boundary) even if an adversarial example has been found.

Findings 2. The effectiveness of input transformations does not necessarily decrease with increase of distortion, since large distortion does not imply stronger adversarial voices.

Findings 2 is based on the comparison between different attacks with the same perturbation budget. To be comprehensive, we also evaluate the effectiveness of transformations on the same attack with different perturbation budgets. We find that the adversarial accuracy drops with the increase of the perturbation budget. This is not surprising since the strength of adversarial voices improves with the increase of the perturbation budget, at the cost of distortion. More details refer to Supplemental Material A.4.2.

Effectiveness versus attack strength. With increase of κ in CW_2 (i.e., attack strength), unsurprisingly, the effectiveness of all the transformations decreases. However, though the attack strength of PGD and CW_∞ attacks increase with #Steps (cf. TABLE 5), the effectiveness of the input transformations (e.g., QT, AT, MS, OPUS, SPEEX and FeCo-o) does *not* decrease monotonically. To understand this, we analyze the strength of adversarial voices before/after applying MS in Fig. 3 and find that the strength of the

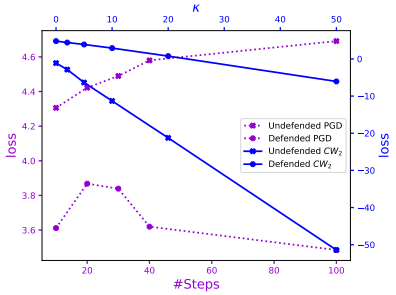


Fig. 3. The loss values (i.e., strength) of the adversarial voices on the model without/with the MS input transformation versus #Steps of PGD and κ of CW_2. The larger the loss of PGD (resp. the smaller the loss of CW_2), the stronger the adversarial examples. The loss of PGD is scaled for better visualization.

adversarial examples crafted by CW_2 remains monotonic after applying MS with increase of κ, while the strength of the adversarial examples crafted by PGD becomes non-monotonic after applying MS with increase of #Steps. It is probably because PGD uses the L_∞ bound while CW_2 does not, hence CW_2 introduces larger distortion with increase of κ, but PGD does not introduce obviously larger distortion with increase of #Steps, as shown in TABLE 5.

Since the step size α may impact the capacity of the PGD attack, we also adopt another three dynamic strategies $\alpha = \frac{\epsilon}{5\times\#\text{Steps}}$, $\alpha = \frac{\epsilon}{\#\text{Steps}}$, and $\alpha = \frac{10\times\epsilon}{\#\text{Steps}}$ which reduces the step size α with increase of #Steps (Recall that previously we set $\alpha = \frac{\epsilon}{5}$). The same phenomenon also occurs (cf. TABLE 9 in Supplemental Material A.4.1), indicating this phenomenon is not due to unsuitable step size.

Findings 3. The effectiveness of input transformations does not necessarily decrease with increase of attack strength.

Overall effectiveness. Transformations are often more effective against L_2 white-box, L_∞ black-box, and signal processing attacks than L_∞ white-box attacks. For instance, AS, LPF, AAC-V, and MP3-V cannot improve any robustness against the PGD and CW_∞ attacks regardless of #Steps, and the CW_2-50 attack. By analyzing the strength of adversarial voices in TABLE 5, we found that:

Findings 4. AS, LPF, AAC-V, and MP3-V are *completely* ineffective against attacks that craft high-confidence adversarial voices (i.e., PGD, CW_∞ and CW_2 with $\kappa = 50$), in non-adaptive setting.

VBR and CBR in speech compression. We noticed significant difference of effectiveness between VBR speech compression (e.g., AAC-V and MP3-V) and CBR speech compression (OPUS, SPEEX, AMR, AAC-C, and MP3-C). For instance, the accuracy of MP3-C (resp. AAC-C) against CW_2-10 is 212 (resp. 11) times larger than that of MP3-V (resp. AAC-V). Compared to CBR speech compression, VBR speech compression dynamically adjusts the bit rate of the voices to better fit to the psychoacoustic perception of the human ear and thus achieves better quality. As a result, although they incur less side effect on the benign voices (A_b of AAC-V and MP3-V only drops by 0% and 0.2% compared to the Baseline), they are limited in disrupting the adversarial perturbation.

Findings 5. VBR speech compression has less side-effect, but are less effective in mitigating adversarial voices.

More findings in the non-adaptive setting refer to Supplemental Material A.4.3.

6 Adaptive Attacks

To evaluate the robustness of transformations in the adaptive setting where the adversary has complete knowledge of defense and attempts to bypass the defense, we design adaptive attacks tailored to input transformations, following the suggestions of [18], i.e., being as simple as possible while resolving any potential optimization difficulties.

To bypass a certain input transformation $g(\cdot)$, the adversary attempts to find an adversarial voice x^{adv} from a benign voice x such that x^{adv} remains adversarial after being transformed by $g(\cdot)$, namely, solving the following optimization problem:

$$\text{argmin}_{x^{adv}} \mathcal{L}(g(x^{adv}), y) \quad \text{such that} \quad \|x^{adv} - x\|_p \leq \epsilon$$

where $\mathcal{L}$ is the loss function used in non-adaptive attack (cross-entropy loss for FGSM, PGD, and margin loss for CW_∞, CW_2, FAKEBOB, and SirenAttack), $p = 2, \infty$ is the L_p norm-based distance, and y is the ground-truth label of x for untargeted attack.

FAKEBOB, SirenAttack, and Kenansville solve the optimization problem without gradient back-propagation, thus can be directly mounted, except that the adaptive version goes through the deployed transformation when querying the model, while the non-adaptive one does not. For differentiable and deterministic transformations (i.e., AS, MS, DS, LPF, and BPF) on which reliable and informative gradients can be computed via back-propagation, the optimization problem can be easily solved by white-box attacks using gradient descents. However, the gradient of the loss function $\mathcal{L}$ w.r.t. x^{adv} cannot be back-propagated for non-differentiable transformations (e.g., QT and speech compressions) while the gradient is less reliable and informative for randomized transformations (e.g., AT and FeCo). To address this issue, we adopt evasion techniques for white-box attacks (i.e., FGSM, PGD, CW_∞, and CW_2 attacks).

6.1 Bypassing W-Transformations

To enable backpropagation of the gradient from a non-differentiable but deterministic W-transformation g, the adversary may utilize Backward Pass Differentiable Approximation (BPDA) [19]. Specifically, during the forward pass, the adversary directly uses g to compute the loss, while uses a differentiable function $\hat{g}$ in the backward pass, i.e., approximating $\nabla_x g(x)$ with $\nabla_x \hat{g}(x)$. We set $\hat{g}(x) = x$, i.e., the identity function, which has been shown effective for breaking non-differentiable input transformations in the image domain [18].

To tackle randomized transformations, the adversary may exploit Expectation over Transformation (EOT) [21], i.e., the loss function is reformulated as $\mathbb{E}_r[\mathcal{L}(g_r(x), y)] \approx \frac{1}{R}\sum_{i=1}^{R} \mathcal{L}(g_{r_i}(x), y)$ where r denotes the randomness of g, r_i is an independent draw of the randomness, and R is the number of independent draws. Intuitively, a randomized transformation is independently sampled multiple times

Algorithm 2 Replicating features

Input: feature matrix $\mathcal{M} = [\mathbf{a}_1, \cdots, \mathbf{a}_N]$; cluster ratio $0 < cl_r < 1$; cluster oracle $\mathcal{O}$ = kmeans or warped-kmeans
Output: replicated feature matrix $\mathcal{M}'$
1: $k \leftarrow \lfloor \frac{1}{cl_r} \rfloor$
2: **for** $(i = 1; i \leq N; i++)$ **do**
  $\mathcal{A}_i \leftarrow$ matrix that replicates the vector $\mathbf{a}_i$ k times
3: **for** $(i = 1; \lceil (N \times k + i - 1) \times cl_r \rceil \neq N; i++)$ **do**
  append the vector $\mathbf{a}_i$ to $\mathcal{A}_i$
4: $\mathcal{M}_1 \leftarrow [\mathcal{A}_1, \cdots, \mathcal{A}_N]$ ▷ concatenate the replicated vectors
5: $[b_1, \cdots, b_{|\mathcal{M}_1|}] \leftarrow \mathcal{O}(\mathcal{M}_1, N)$
6: Let $i_1, \cdots, i_N$ be a permutation of $1, \cdots, N$ s.t. for each $1 \leq j \leq N$, most of vectors of $\mathcal{A}_{i_j}$ are divided into the b_{i_j}-cluster
7: $\mathcal{M}' \leftarrow [\mathcal{A}_{i_1}, \cdots, \mathcal{A}_{i_N}]$
8: **return** $\mathcal{M}'$

and the average of the loss function is used during gradient descent. It reduces the variance of the gradient and enables a more stable search direction. We remark that four differentiable and randomized transformation based defenses have been broken using EOT in the image domain [18], [19].

6.2 Bypassing F-Transformations

Since FeCo is differentiable and randomized, one could use EOT to bypass FeCo (cf. Section 6.1). Below, we design more specific evasion techniques for white-box attacks, tailored to FeCo, called Replicate attack, including Replicate-F(feature) and Replicate-W(ave).

Replicate-F. To bypass FeCo, the adversary first crafts an adversarial voice x' on the model *without* FeCo, and then builds a new feature matrix $\mathcal{M}'$ from the feature matrix $\mathcal{M}$ of x' with the goal FeCo$(\mathcal{M}') = \mathcal{M}$, i.e., when $\mathcal{M}'$ is fed to the model defended by FeCo, $\mathcal{M}'$ is likely compressed to $\mathcal{M}$, leading to a successful attack.

The desired feature matrix $\mathcal{M}'$ is built by applying Algorithm 2. Suppose $\mathcal{M} = [\mathbf{a}_1, \cdots, \mathbf{a}_N]$ where $\mathbf{a}_i$ is the feature vector of the i-th frame. It first replicates each feature vector $\mathbf{a}_i$ of $\mathcal{M}$ by $k = \lfloor \frac{1}{cl_r} \rfloor$ times and then appends vectors to the replicated vectors $\mathcal{A}_i$'s until the concatenated matrix $\mathcal{M}_1$ of $[\mathcal{A}_1, \cdots, \mathcal{A}_N]$ will lead to a feature matrix with N frames after applying FeCo. It is expected that FeCo$(\mathcal{M}_1)$ has the same frames as $\mathcal{M}$. However, the order of frames of FeCo$(\mathcal{M}_1)$ may differ from that of $\mathcal{M}$. To overcome this problem, we run the clustering algorithm with the parameter cl_r on the matrix $\mathcal{M}_1$ to get the order of the frames of FeCo$(\mathcal{M}_1)$. This order is used to permute the replicated vectors $\mathcal{A}_i$'s intended to make FeCo$(\mathcal{M}') = [\frac{\sum \mathcal{A}_{i_1}}{|\mathcal{A}_{i_1}|}, \cdots, \frac{\sum \mathcal{A}_{i_N}}{|\mathcal{A}_{i_N}|}]$ being $\mathcal{M}$.

Replicate-W. Replicate-F is infeasible in practice, as exposed APIs *only* accept waveforms. Thus, we introduce Replicate-W, which is similar to Replicate-F except that the adversarial voice x^{adv} is reconstructed from $\mathcal{M}'$ using Griffin-Lim algorithm [64] and fed to SRS defended with FeCo.

7 Evaluation of Adaptive Attacks

7.1 Evaluation Setup

We evaluate transformations in the same setup as in Section 5 against adaptive attacks derived from a subset of representative attacks according to Section 6. For adaptive attacks derived from FGSM, CW_2-0, FAKEBOB, SirenAttack and Kenansville, we consider all the transformations, as they are effective in the non-adaptive setting, but the effectiveness varies. For adaptive attacks derived from PGD-10, PGD-100, CW_∞-10, and CW_∞-100, we do not consider AS, DS, LPF, and BPF, as they are differentiable, deterministic, and almost completely ineffective in the non-adaptive setting. The CW_2-2 (resp. CW_2-50) attack is considered only when a transformation is effective (i.e., at least 5% accuracy) against CW_2-0 (resp. CW_2-2). We do not consider all the combinations of attacks and transformations, as the current experiments already require substantial effort.

7.2 Results

The results are shown in TABLE 6. Overall, the effectiveness varies with transformations and attacks. Below, we compare the results with those obtained in the non-adaptive setting (i.e., TABLE 4), by distinguishing if the transformations are differentiable or not.

Results of non-differentiable transformations (gray color in TABLE 6). First, QT becomes less effective against both white-box and black-box attacks, indicating both BPDA and adaptive black-box attacks are able to circumvent QT.

Second, against adaptive white-box attacks, the effectiveness of CBR speech compressions (i.e., OPUS, SPEEX, AMR, AAC/MP3-C) does not decrease, indicating that BPDA is not able to circumvent them. Indeed, (1) BPDA cannot reduce the accuracy of speech CBR compressions on the adversarial examples crafted by FGSM, PGD, and CW_∞-0 when compared with the results in TABLE 4. (2) Though BPDA can reduce the accuracy on the adversarial examples crafted by CW_2-0 and CW_2-2, much more distortions are introduced than the non-adaptive CW_2 attack, e.g., the SNR of the adaptive CW_2-0 (with BPDA) on AAC-C (resp. MP3-C) is 32.67 dB (resp. 34.70 dB), 20 dB (resp. 18 dB) smaller than that of the non-adaptive CW_2-0 (52.99 dB, cf. TABLE 5). Recall that CW_2 does not have any perturbation threshold, while other attacks have. Thus, adaptive CW_2 attacks still achieve high attack success rate at the cost of distortion.

In contrast, we found that BPDA with the identity function is effective in breaking VBR speech compression (i.e., AAC/MP3-V). Compared with the result of non-adaptive CW_2-0 attack in TABLE 4, the adaptive CW_2-0 attack equipped with BPDA reduces the accuracy of AAC-V (resp. MP3-V) by 70.1% (resp. 59.8%) with no more than 0.2 and 4.1 dB decrease in PESQ and SNR, respectively.

To understand why BPDA has different effectiveness between QT, CBR and VBR speech compressions, we checked the appropriateness of approximating non-differentiable transformations by the identity function and found that QT and VBR speech compressions are much closer to the identity function than CBR speech compressions (cf. Supplemental Material A.5), indicating that BPDA with the identity function is not strong enough to bypass CBR speech compressions, and better approximation functions are required to circumvent them. We leave this as future work (cf. Section 9.1 for discussion).

Findings 6. BPDA with identity function can evade non-differentiable QT and VBR speech compressions, but fail to evade CBR speech compressions.

TABLE 6: Results (A_a, SNR, PESQ) of transformations against adaptive attacks

Defense	Adaptive Techniques	L_∞ white-box attacks					L_2 white-box attacks									black-box attacks		
		FGSM	PGD-10	PGD-100	CW_∞-10	CW_∞-100	CW_2-0			CW_2-2			CW_2-50			FAKEBOB	SirenAttack	Kenansville
		A_a	A_a	A_a	A_a	A_a	A_a	SNR	PESQ	A_a	SNR	PESQ	A_a	SNR	PESQ	A_a	A_a	A_a
QT	BPDA	18.6%	0%	0%	0%	0%	14.6%	46.81	3.86	0%	44.04	3.71	-			40.1%	75.0%	9.9%
AT	EOT	18.7%	4.3%	1.8%	4.5%	1.9%	64.4%	37.47	3.03	26.2%	35.45	2.88	0%	20.71	1.70	96.67%	95.0%	18.5%
AS	✗	31.5%	-	-	-	-	19.0%	49.70	4.16	0%	48.49	4.11	-			14.5%	93.0%	9.8%
MS	✗	1.6%	0%	0%	0%	0%	4.7%	61.76	4.45	-			-			0.3%	23.0%	6.5%
DS	✗	24.2%	-	-	-	-	18.2%	57.28	4.35	0%	55.02	4.29	-			15.0%	93.0%	8.5%
LPF	✗	32.6%	-	-	-	-	20.2%	55.34	4.35	0%	53.46	4.29	-			18.8%	95.9%	7.1%
BPF	✗	26.4%	-	-	-	-	17.3%	57.98	4.37	0%	55.99	4.31	-			12.3%	82.7%	6.8%
OPUS	BPDA	89.1%	86.8%	84.4%	86.5%	84.0%	25.1%	20.97	1.89	0%	15.94	1.71	-			82.3%	73.2%	8.7%
SPEEX	BPDA	89.7%	80.6%	75.4%	80.0%	75.2%	1.9%	24.33	1.92	-			-			87.7%	72.0%	7.2%
AMR	BPDA	90.4%	73.2%	63.4%	73.5%	63.5%	2.1%	24.30	1.96	-			-			92.0%	80.1%	6.3%
AAC-V	BPDA	51.9%	0%	0%	0%	0%	2.3%	48.96	4.06	-			-			44.9%	97.0%	9.1%
AAC-C	BPDA	88.8%	43.2%	6.2%	44.5%	6.7%	19.9%	32.67	2.59	0%	29.23	2.36	-			23.1%	65.0%	8.3%
MP3-V	BPDA	52.2%	0%	0%	0%	0%	2.4%	49.95	4.12	-			-			46.4%	96.1%	6.9%
MP3-C	BPDA	89.4%	10.2%	0.9%	10.5%	1.2%	15.5%	34.70	2.88	0%	31.11	2.64	-			54.2%	64.2%	7.3%
FeCo-o(k)	EOT	54.1%	0%	0%	0%	0%	90.4%	56.20	4.14	88.0%	53.54	4.05	1.2%	18.38	1.57	92.17%	96.4%	31.0%
	Replicate-W	68.0%	39.4%	49.0%	39.3%	49.9%	82.7%	-	-	78.7%	-	-	58.6%	-	-	87.8%	83.9%	20.0%
	Replicate-F	72.4%	7.9%	15.6%	7.3%	14.5%	92.8%	-	-	88.6%	-	-	36.7%	-	-	98.1%	93.2%	22.6%

Note: The accuracy in red indicates that an adaptive attack is not stronger than its non-adaptive version. The cells with gray (resp. green) color indicate that the transformations are non-differentiable (resp. randomized). Distortion levels of L_∞ attacks are not reported since they are similar. The distortion levels of Replicate attacks are not reported since the benign and adversarial voices do not align with each other due to the replication operation.

We highlight that in the image domain, [19] and [18] successfully evade all the seven input transformation-based adversarial defenses using BPDA with the identity function, which is inconsistent with our Findings 6. Also, while [65] showed MP3 robust audio adversarial examples against speech recognition models can be crafted with BPDA at the cost of approximately 15dB larger distortion (close to our result of MP3-C), Findings 6 shows that MP3-V can be easily evaded with BPDA without obvious distortion increase.

Third, CBR speech compressions become less effective against adaptive FAKEBOB and SirenAttack, especially, AAC-C and MP3-C reduce 53.3% and 16.90% accuracy against adaptive FAKEBOB, respectively. However, AAC/MP3-V achieve higher accuracy, indicating that adaptive FAKEBOB and SirenAttack are limited in circumventing VBR speech compressions. It is because the gradients estimated by NES of FAKEBOB for AAC/MP3-V are not informative enough, and the particles moving direction of PSO in SirenAttack is not stable, due to the variable bit rate of AAC/MP3-V.

Findings 7. Variable bit rate (VBR) makes speech compressions more resistant against adaptive black-box attacks.

Results of differentiable transformations (non-gray color in TABLE 6). All the deterministic transformations become less effective against white-box and black-box adaptive attacks, except for AS, DS, LPF, and BPF against SirenAttack because the perturbation budget $\epsilon = 0.002$ is not sufficient enough for SirenAttack to evade these transformations. When $\epsilon = 0.02$, the adaptive SirenAttack becomes stronger than the non-adaptive one, reducing at least 16% accuracy, on these transformations (cf. Supplemental Material A.6).

Randomized transformations (i.e., AT and FeCo-o(k)) can also be evaded by the white-box adaptive attacks with EOT or larger parameter κ. However, AT and FeCo-o(k) remain effective on the adversarial examples crafted by the black-box adaptive attacks FAKEBOB, SirenAttack, and Kenansville (except for AT due to the larger distortion introduced by Kenansville which suffices to overcome the randomness of AT). This is because: their randomness makes the estimated gradients of NES uninformative for FAKEBOB, the moving direction of PSO unreliable for SirenAttack, and randomized decision for Kenansville.

Findings 8. Differentiable transformations become less effective against the white-box adaptive attacks, but randomized transformations remain resistant to the black-box adaptive attacks.

Replicate attack versus EOT. We observe that EOT is more effective than the Replicate attack to bypass FeCo-o(k). To understand the reason, we analyze if the expectation (i.e., $\text{FeCo}(\mathcal{M}') = \mathcal{M}$) of the Replicate attack is satisfied. We found that $\text{FeCo}(\mathcal{M}')$ has almost the same frames (i.e., feature vectors) as $\mathcal{M}$, but their orders are not the same, due to the randomness of FeCo. Indeed, it is impossible to ensure the same orders, even if a brute-force adversary can enumerate the randomness, where the adversary has to craft and submit an adversarial voice for each randomness, would result in a low success rate (cf. Supplemental Material A.7). In contrast, EOT allows to craft an adversarial voice that remains adversarial against the randomness of FeCo by taking average of the loss functions conditioned at multiple randomness during the gradient descent.

Besides, Replicate attack replicates the speech content of each frame, and the lossy reconstruction of voices from features introduce additional noise, making the adversarial voices more perceptible (visit our website for listening audios) and less robust (i.e., Replicate-W is worse than Replicate-F for strong attacks).

Findings 9. Against FeCo, EOT is more effective than Replicate attack in terms of both attack success rate and imperceptibility.

8 Evaluation of Transformations on Adversarially Trained Model

8.1 Evaluation Setup

As ivector-PLDA cannot be adversarially trained due to unsupervised learning, we adversarially train AudioNet for the CSI-NE task using the datasets Spk_{251}-train and Spk_{251}-test for training and testing, respectively. The training uses a minibatch of size 128 for 300 epoches, cross-entropy loss as the objective function, and Adam [66] to optimize trainable parameters. The naturally trained model is denoted by Standard. For adversarial training, we use PGD with 10

TABLE 7: Results (A_a, SNR, PESQ) on Standard, Vanilla-AdvT, and AdvT+Transformation

	R1 Score	A_b	L_∞ white-box attacks					L_2 white-box attacks			black-box attacks		
			FGSM	PGD-10	PGD-100	CW_∞-10	CW_∞-100	CW_2-1			FAKEBOB	SirenAttack	Kenansville
			A_a	A_a	A_a	A_a	A_a	A_a	SNR	PESQ	A_a	A_a	A_a
Standard	6.54	99.06%	19.61%	0%	0%	0%	0%	0%	55.87	4.47	0.35%	0.38%	0.03%
Vanilla-AdvT	61.48	95.67%	75.20%	58.19%	53.83%	58.95%	55.56%	0%	36.96	3.91	85.63%	86.73%	0.03%
AdvT+QT	67.68	95.74%	88.19%	72.12%	64.08%	73.20%	65.43%	0.7%	46.59	3.86	79.84%	88.81%	0.31%
AdvT+AT	71.11	95.57%	71.10%	61.10%	59.22%	61.47%	59.89%	9.3%	36.21	3.90	94.69%	95.39%	39.80%
AdvT+AS	58.35	93.59%	82.72%	53.83%	43.12%	54.10%	45.24%	0%	35.46	3.45	83.55%	87.08%	0.03%
AdvT+MS	54.66	92.76%	65.85%	49.77%	44.13%	50.33%	46.66%	0%	37.85	3.66	76.38%	77.24%	0.17%
AdvT+DS	56.41	95.32%	70.14%	51.44%	44.06%	52.13%	45.41%	0%	36.23	3.91	79.91%	85.04%	0.69%
AdvT+FeCo-o(k)	**88.03**	**97.81%**	**95.06%**	**93.65%**	**85.50%**	**94.14%**	**86.11%**	**96.0%**	**29.89**	**2.53**	**98.08%**	**97.42%**	**39.94%**

Note: The top-1 is highlighted in blue excluding Standard. The results in green background indicate that the transformation worsens adversarial training.

steps (i.e., PGD-10) to generate adversarial examples. The model is denoted by Vanilla-AdvT.

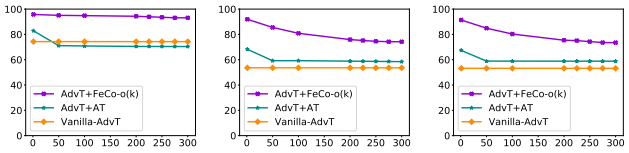
For each chosen transformation X, we implement it as a proper layer in AudioNet. Note that this layer does not involve any trainable parameter. The resulting network is adversarially trained the same as above, except that BPDA is leveraged for training the network with non-differentiable transformations and EOT with R=10 is leveraged for training the network with randomized transformations. The resulting model is denoted by AdvT+X. We do not consider speech compressions, LPF and BPF, as BPDA is not effective for estimating the gradients of speech compressions, and the accuracy of the resulting model with LPF/BPF is extreme low on both training dataset (i.e., 24.10%/23.65%) and testing dataset (i.e., 2.04%/2.25%).

The adaptive attacks are derived from FGSM, PGD-10, PGD-100, CW_∞-10, CW_∞-100, CW_2-1, FAKEBOB, SirenAttack, and Kenansville, armed with EOT (R=50) and BPDA to evade randomized and non-differentiable transformations. To improve the attack capability of FAKEBOB, we increase the parameter samples_per_draw m to 300, allowing more precise gradient estimation at the cost of increased attack overhead. Since adversarially trained models tend to yield smaller loss than naturally trained one, we increase the initial trade-off constant c of CW_2 attack from 0.001 to 0.1 when attacking Vanilla-AdvT and AdvT+X. This helps finding adversarial examples with better imperceptibility according to our experiments.

8.2 Results

The results are reported in TABLE 7. We observe that the sole adversarial training (i.e., Vanilla-AdvT) is effective for defeating adversarial examples compared over Standard except for Kenansville, at the cost of slightly sacrificing accuracy on benign examples (i.e., A_b reduces from 99.06% to 95.67%). Adversarial training either significantly improves the accuracy by more than 53% on the adversarial examples crafted by L_∞ attacks, or amplifies the distortions of the adversarial examples crafted by CW_2-1 (the SNR of Vanilla-AdvT is 18 dB smaller than that of Standard). However, adversarial training does not improve the model accuracy on the adversarial examples crafted by Kenansville. This is not surprising since Kenansville is a signal processing-based attack while the adversarial examples used for adversarial training is generated by the optimization-based attack PGD-10. We also tried to improve the model robustness against Kenansville by incorporating Kenansville in adversarial training, but the result is not promising (cf. Section 9.1 for discussion).

While sole adversarial training is often effective compared over Standard, the combination of adversarial training with a transformation, highlighted in green color in TABLE 7, does not necessarily bring the best of both worlds, which also exists in image domain [18].



(a) #Steps=1 (b) #Steps=100 (c) #Steps=200

Fig. 4. x-axis is EOT_size (R) and y-axis is A_a.

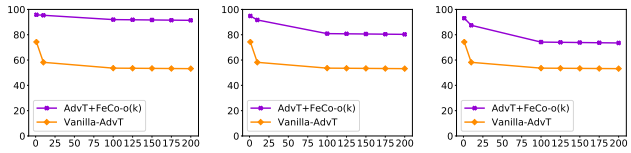
Interestingly, we found that adversarial training combined with FeCo-o(k), i.e., AdvT+FeCo-o(k), is very effective, achieving higher accuracy on both the adversarial and benign examples compared with Vanilla-AdvT. This improvement is brought by the randomness of FeCo. In fact, during the training of AdvT+FeCo-o(k), the training data are randomly transformed by FeCo, which enhances the quantity and diversity of the training data, similar to data augmentation. Consequently, the distribution mimicked by the training dataset $\{(x_i, y_i)\}_{i=1}^{B}$ becomes closer to the underlying data distribution $\mathcal{D}$ (cf. Section 3.3), on which AdvT+FeCo-o(k) encounters more diverse adversarial examples during training. Thus, it becomes more robust than Vanilla-AdvT. A similar result is also reported in the image domain [33], where some image data augmentation methods improve adversarial robustness.

Compared to the other transformations, FeCo enjoys larger randomness space than AT (cf. Section 8.3) and other deterministic transformations (without randomness), hence AdvT+FeCo-o(k) outperforms other AdvT+X.

8.3 Attack Parameters Tuning

To thoroughly evaluate the robustness of AdvT+FeCo-o(k) against adaptive versions of the PGD and CW_2 attacks, we further conduct a series of experiments by tuning the attack parameters, including EOT_size (R), number of steps (#Steps), step_size (α), and confidence (κ). Since these experiments on the entire Spk_{251}-test dataset require huge effort, we randomly select 1,000 voices out of 2,887 voices in Spk_{251}-test from which adversarial examples are crafted.

EOT_size (R). We study the impact of EOT_size (R) on the effectiveness of AdvT+FeCo-o(k). We set PGD's step_size $\alpha = \epsilon/5 = 0.0004$ (the same as previous experiments) and #Steps=1, 100, 200. For each number of steps (#Steps), EOT_size (R) ranges from 1 to 300. The results are shown in Fig. 4. We observe that with the increase of EOT_size (R), the accuracy of both AdvT+FeCo-o(k) and AdvT+AT decreases. This is because larger EOT_size (R) allows EOT to more accurately approximate the distributions of randomized trans-



(a) r=1 (b) r=100 (c) r=300

Fig. 5. x-axis is the number of steps (#Steps), and y-axis is A_a, where #Steps=1 is indeed the FGSM attack.

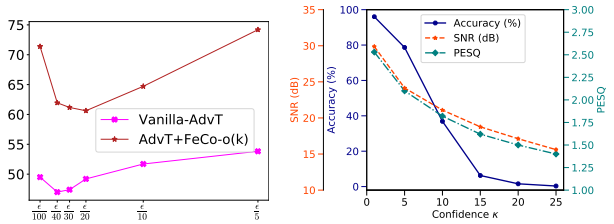
formations, enabling the PGD attack to obtain more reliable gradient and thus more stable search direction for adversarial examples. However, when $R \geq 275$ (resp. $R \geq 50$), further increasing R has negligible effect on AdvT+FeCo-o(k) (resp. AdvT+AT), i.e., the accuracy becomes stable. Note that AdvT+FeCo-o(k) converges at a larger EOT_size (R) than AdvT+AT, i.e., 275 vs. 50. Recall that EOT is exploited to overcome the randomness of a transformation. Thus, EOT_size (R) is a reasonable metric for quantifying the degree of randomness that a transformation introduces. Accordingly, we can conclude that FeCo introduces larger randomness than AT.

Number of steps (#Steps). We study the impact of the number of steps (#Steps) in the PGD attack on the effectiveness of AdvT+FeCo-o(k). We set PGD's step_size $\alpha = \epsilon/5 = 0.0004$ and EOT_size $R = 1, 100, 300$. The number of steps (#Steps) ranges from 1 to 200 for every EOT_size (R). The results are shown in Fig. 5. We observe that the accuracy of AdvT+FeCo-o(k) decreases gradually when #Steps increase from 1 to 100. This is not surprising as increasing #Steps improves the strength of adversarial examples (cf. Fig. 3). However, when #Steps>100, the accuracy of AdvT+FeCo-o(k) remains almost unchanged with the increase of the number of steps (#Steps).

Step_size (α). Based on the above results, we fix #Steps=100 and EOT_size $R = 275$ when studying the impact of step_size (α) on the effectiveness of AdvT+FeCo-o(k) by setting $\alpha = \epsilon/100$, $\epsilon/40$, $\epsilon/30$, $\epsilon/20$, $\epsilon/10$, $\epsilon/5$. The results are shown in Fig. 6(a). We found that decreasing step_size reduces the accuracy of both Vanilla-AdvT and AdvT+FeCo-o(k). We conjecture that the PGD attack with small step_size is less likely to oscillate across different directions, thus can search for adversarial examples in a more stable way. However, when $\alpha \leq \epsilon/20$ (resp. $\alpha \leq \epsilon/40$), decreasing step_size (α) reduces the attack success rate on AdvT+FeCo-o(k) (resp. Vanilla-AdvT).

From the above three stuides, we can observe that the accuracy of AdvT+FeCo-o(k) plateaus at 60.62% with $R = 275$, #Steps=100, and $\alpha = \epsilon/20$, while the accuracy of Vanilla-AdvT plateaus at 47.0% with $R = 1$, #Steps=100, and $\alpha = \epsilon/40$. Thus, AdvT+FeCo-o(k) achieves 13.62% higher accuracy than Vanilla-AdvT. Furthermore, the attack has to query the AdvT+FeCo-o(k) model $275 \times 100 = 27,500$ times, while it only has to query the Vanilla-AdvT model $1 \times 100 = 100$ times. This indicates that FeCo-o(k) significantly improves the attack cost by two orders of magnitude.

Confidence (κ). We launch the CW_2 attack by setting the parameter $\kappa = 1, 5, 10, 15, 20, 25$, where the larger κ, the stronger the attack. As shown in Fig. 6(b), though the accuracy on the adversarial examples decreases with the increase of κ, the distortion also increases. For instance, when $\kappa = 25$, the attack success rate is nearly 100%, but



(a) x-axis is step_size (α) and y-axis is A_a. (b) Accuracy and distortion of adversarial voices with the increase of κ for CW_2.

Fig. 6. Tuning the step_size (α) and confidence (κ).

the SNR (resp. PESQ) is 15.61 dB (resp. 1.40), 40.26 dB (resp. 3.07) smaller than that of Standard, indicating that the adversarial examples become much less imperceptible. This demonstrates the effectiveness of AdvT+FeCo-o(k) against powerful attacks.

Findings 10. Among the adversarially trained models combined with transformations, AdvT+FeCo-o(k) is the unique one that is effective against all the adaptive attacks. Compared with Vanilla-AdvT, it improves the accuracy on both benign examples and adversarial examples against L_∞, L_2 and signal processing-based adaptive attacks, largely increases the attack cost of the PGD based adaptive attack, and significantly worsens the imperceptibility of adversarial examples crafted by the CW_2 based adaptive attack.

9 Discussion

We discuss some key findings and the limitations of our study, interspersed with possible future works motivated by them.

9.1 Discussion of Findings

Combination of different transformations. According to Findings 1, TABLE 4, and TABLE 6, the effectiveness of transformations varies with attacks. Moreover, different types of transformations operate on different domains (time vs. frequency), different levels (waveform vs. acoustic feature) and own different properties (differentiable vs. non-differentiable, deterministic vs. randomized). Therefore, it is interesting to study if the combinations of transformations (e.g., AT and FeCo) could improve adversarial robustness.

Attacks against speech compression defenses. Findings 6 and Findings 7 reveal that BPDA, FAKEBOB and SirenAttack are hard to circumvent non-differentiable CBR and VBR speech compression, respectively. BPDA cannot succeed since replacing speech compression with the identity function in the backward pass is not precise enough (cf. Fig. 10 in Supplemental Material). Diving deeper into speech compression, we found that its bit allocation would assign unequal number of bits to voice sample points, according to their contribution to human perception of the voices. Consequently, the transformed voice by speech compression does not align with the original one in time axis, making speech compression far from the identity function. To improve BPDA, we may utilize time sequence alignment techniques, e.g., dynamic time warping [67], to align the original and

transformed voices to make speech compression close to the identity function as much as possible. Another potential solution is to design more accurate approximation functions than the identity function, e.g., differentiable Variational AutoEncoder [68] with the origin voice and transformed voice as the input and latent variables, respectively. The AutoEncoder is first trained to learn the mapping from origin voices to transformed voices and then utilized to replace the non-differentiable speech compressions in the backward pass. The failure of FAKEBOB and SireAttack may be attributed to the large non-smoothness introduced by the variable bit rate of speech compression. The smoothness assumption of NES and PSO does not hold anymore [69], making the estimated gradient of NES and the search direction of PSO not reliable and informative enough for gradient descent. $\mathcal{N}$ATTACK [69], which will not be impeded by the non-smoothness of models, and gradient-free decision-only attacks from the image domain, e.g., evolutionary attack [70], may be good alternatives to evade speech compression.

Black-box attacks against randomized defenses. According to Findings 8, all the black-box attacks (FAKEBOB, SirenAttack, and Kenansville) have limited attack success rate on the models with randomized transformations (e.g., AT and FeCo). This is probably because NES of FAKEBOB becomes ineffective for estimating gradients, PSO of SirenAttack becomes unstable for searching better particle locations, and Kenansville gets misled in updating the attack factor, in presence of randomness. To bypass such randomized transformations, one may use $\mathcal{N}$ATTACK which is effective in breaking the randomized defenses in the image domain. Adapting $\mathcal{N}$ATTACK to speaker recognition is an interesting future work.

Robust training against Kenansville. The results in TABLE 7 show that adversarial training fails to improve robustness against Kenansville. The reason is that the adversarial training uses the optimization-based attack PGD, while Kenansville is a signal processing-based attack. We also tried to incorporate Kenansville into adversarial training but found that it not only fails to increase adversarial robustness against Kenansville, but also significantly degrades accuracy on benign voices. The former may be due to low-confidence of adversarial examples crafted by Kenansville that are not suitable for solving the inner maximization problem in adversarial training (cf. in Section 3.3) while the latter may be due to large distortion introduced by Kenansville. Details refer to TABLE 5. Since adversarial training does not work well for Kenansville, we may turn to other robustness training techniques, e.g., Lipschitz regularization [6]. In addition, Kenansville can be defeated by liveness detection [71], [72] when it is launched over the air. Liveness detection detects over-the-air attacks by exploiting the different characteristics of the voices generated by human vocal tract and electronic loudspeaker, so it can defend against both optimization-based and signal-processing-based over-the-air attacks.

9.2 Discussion of Limitations

Threats to Validity. In this study, we adopt ivector-PLDA and AudioNet as the speaker recognition models, and four datasets derived from Librispeech as the datasets. It is not clear whether the findings based on them can be extended to other models and datasets. As a first attempt for confirmation, we choose another deep learning-based model DeepSpeaker [27], which was released by Baidu Inc. and is one of the state-of-the-art speaker recognition models, and another dataset VoxCeleb [73] which has different speakers, utterances, and subjects background (e.g., ethnicities, accents, age, and profession) from Librispeech. We re-perform part of experiments on them, and detailed experimental settings as well as results refer to Supplemental Material A.9, from which we observe that the related findings still hold. However, there are still many models and datasets that we cannot cover one by one, e.g., wav2vec 2.0 [74], [75], which stores speaker information of waveforms into the representations of silent segments [76], and LibriTTS [77], due to the huge cost.

Suitability of audio imperceptibility metrics. We use L_∞ and L_2 norms to quantify the perturbation magnitude in adversarial example generation, and adopt SNR and PESQ to measure the imperceptibility of crafted adversarial voices. These metrics have been widely adopted in the literature [6], [7], [8], [10], [11], [12], [13] and in general, can consistently reflect the degree of distortions according to our experimental results. Moreover, PESQ is an objective perceptual measure simulating the human auditory system [62]. However, it remains unknown to what extent do these metrics correlate with human hearing perception. In the image domain, the proximity of two images measured by L_p norm is neither necessary nor sufficient for them to be visually indistinguishable by humans [78]. Therefore, it is worthy to explore in future the sufficiency and necessity of these metrics in quantifying the audio perceptual similarity.

Securing commercial SRSs. We did not directly target commercial SRSs, although they are also vulnerable to black-box attacks [12], [79]. The reason is that it is more important to consider the most powerful adversaries when evaluating defenses, while the adversaries are not able to mount white-box attacks without having access to the internal structures of commercial SRSs. Instead, we directly evaluate defenses against the black-box attacks FAKEBOB [12], SirenAttack [13] and Kenansville [15] which could be used to attack commercial SRSs and FAKEBOB is able to fool commercial SRSs. Investigating and evaluating if our findings are applicable to commercial SRSs is left for future work.

Detection of adversarial voices. While we focus on adversarial training and transformation based defenses against adversarial attacks, effective transformations could be leveraged to detect adversarial voices by comparing the degree-of-change of benign and adversarial voices before and after transformations [32]. This is reasonable as benign voices are generally more robust [80], their results are less likely to change after transformations, which is validated by our Findings 11 in Supplemental Material A.4.3.

Defending against over-the-air attacks. Our evaluation focuses on digital attacks where adversarial voices are directly fed to the SRS via exposed API, as it is more important to evaluate defenses against powerful adversaries while over-the-air attack will be compromised by various sources of distortions [53]. We emphasize that input transformations are also applicable to over-the-air attacks where the adversarial voices are played and recorded by hardware

and transmitted in the air. Transformations can back-up liveness detection [71], [72] when liveness detection has false negatives, where liveness detection detects over-the-air attacks by exploiting the different characteristics of the voices generated by human vocal tract and electronic loudspeaker. Evaluating the effectiveness of these transformations in defending against over-the-air attacks is left for future work.

Input transformations against other attacks. This work focuses on defending against adversarial attacks. There are other attacks against SRSs which have different attack goals and scenarios from adversarial attack. Thus, it is interesting to investigate whether input transformations can defend against those attacks. As a first attempt, we carry out a preliminary evaluation against hidden voice attack [81] and speech synthesis attack [82] (cf. Supplemental Material A.8). We found that input transformations are also effective in mitigating these two attacks and speech synthesis attack is more difficult to defeat than the other two attacks. More thorough evaluations against more other attacks are needed in the future.

10 Related Work

Adversarial attacks and defenses in the speech and speaker recognition domains recently have attracted intensive attention. Though both of them share a similar feature extraction pipeline, they perform different tasks and speaker recognition owns unique enrollment phase and decision making mechanism [12], [83]. Thus, in this section, we do not discuss adversarial attacks and defenses that focus on speech recognition [31], [40], [65], [84], [85], [86], [87], [88] (cf. [63], [83] for survey). There are other voice attacks in the speaker recognition domain, such as hidden voice attacks [81] and spoofing attacks [82], [89]. Though these attacks have different attack goals and scenarios from adversarial attacks [12], our preliminary evaluation shows that it is possible to mitigate hidden voice attack [81] and speech synthesis attack [82] via input transformations. Below, we discuss adversarial attacks and defenses in the speaker recognition domain.

Adversarial attacks. Existing white-box attacks in the speaker recognition domain are derived from the attacks in the image recognition domain. The FGSM method was adopted to attack the CSI-NE task [14] and the SV task [4], [5]. Zhang at al. used PGD to attack the CSI-NE task [7]. Jati at al. attacked the CSI-NE task by leveraging FGSM, PGD, CW_∞ and CW_2 [6] methods. However, these attacks have not been thoroughly evaluated on the systems with various defenses and it is difficult to conclude which one is better due to inconsistent benchmarks (e.g., models and datasets). We consider all these white-box attacks and adaptive variants thereof in this work. Though our main goal is to investigate and evaluate transformation and adversarial training based defenses, our results also provide a fair comparison of these attacks under the same settings when various defenses are deployed.

There are also some specific white-box attacks, aiming at crafting universal perturbations [8], [9], [90] or improving the imperceptibility of adversarial voices [10], [11], yet these works did not consider any defense. Since the essential optimization framework of these attacks is the same as the attacks considered in this work, we do not incorporate these attacks into our study.

FAKEBOB [12], SirenAttack [13], Kenansville [15], and Occam [79] are four black-box adversarial attacks targeting SRSs, where FAKEBOB, SirenAttack, and Occam are optimization-based attacks, and Kenansville is a signal processing-based attack. All of them, except for Occam which is not publicly available and non-trivial to reproduce, have been used to evaluate defenses in this work.

Adversarial defenses: mitigation and detection. Robust training is one way to mitigate adversarial examples. [6], [13] showed that adversarial training can enhance the robustness of models. [6] also proposed another technique which adds a regularization term using Lipschitz smoothness to the loss function for model training. This technique performs better than FGSM based adversarial training, but worse than PGD based adversarial training. This motivated us to evaluate PGD based adversarial training in this work.

The transformations (QT, MS and DS) and (DS and AS) have been evaluated against FAKEBOB and SirenAttack respectively. But, they were neither combined with adversarial training nor thoroughly evaluated under various attacks. Our evaluation shows that these transformations are *not* effective against adaptive attacks and *cannot* improve the adversarial robustness of adversarially trained models. Furthermore, we investigate and evaluate significantly more defenses against both non-adaptive and adaptive attacks. We note that AT, AutoEncoder [80], and GAN [91] have been evaluated against four white-black attacks in [92]. Compared to the transformations considered in this work, AutoEncoder and GAN are data-dependent methods which require additional overhead for training from benign examples to model the distribution of unperturbed voices, thus may exhibit different performance on difference datasets. Although BPDA was used to solve the non-differentiability of GAN in [92], the randomness of AT was not properly addressed, leading to false sense of adversarial robustness. Our findings show that AT becomes ineffective against adaptive attack armed with EOT to address the randomness. Moreover, [92] did not consider black-box attacks, while we did and found some useful related findings (Findings 6-8).

Detection is another way to defend against adversarial voices. [93] proposed to detect adversarial examples by training a CNN-based binary classifier, while [94] checks the consistence of results of twin models. However, these approaches have not been evaluated against adaptive attacks and may be evaded by incorporating the detector into loss functions [95]. Another direction is liveness detection [71], [72] which detects malicious audios by exploiting the different characteristics of the voices generated by human vocal tract and electronic loudspeaker. Liveness detection is a promising approach for defeating physical adversarial attacks. However, it is not suitable for API attacks where adversarial voices are directly fed to the SRSs in the form of audio file via exposed API.

11 Conclusion

We have systematically investigated diverse transformations for mitigating adversarial voices in the speaker recognition domain, including waveform-level transformations

in both time-domain and frequency-domain, speech compression, and feature-level transformations, and covering all the differentiable, non-differentiable, deterministic, and randomized types. We have thoroughly evaluated those transformations on both naturally trained and adversarially trained models against promising white-box and black-box attacks, as well as carefully designed adaptive variants for circumventing different types of transformations. Our study revealed lots of interesting and useful findings for both researchers and practitioners.

Among all the transformations, we showed that our novel feature-level transformation FeCo is rather effective against black-box attacks and improves the robustness of adversarially trained models against both white-box and black-box adaptive attacks in terms of accuracy, attack cost, and distortion level. This opens up a new research direction on transformations for mitigating adversarial examples. We pointed out many possible future works in both adversarial attacks and defenses in the speaker recognition domain, and released our evaluation platform SPEAKERGUARD to foster further research.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant No. 62072309, the Ant Group, and the CAS Project for Young Scientists in Basic Research under Grant No.YSBR-040.

REFERENCES

[1] "Kaldi toolkit," https://github.com/kaldi-asr/kaldi, 2022.
[2] "Microsoft azure speaker recognition," https://azure.microsoft.com/en-us/services/cognitive-services/speaker-recognition, 2022.
[3] TD Bank voiceprint, https://www.tdbank.com/bank/tdvoiceprint.html, 2022.
[4] F. Kreuk, Y. Adi, M. Cissé, and J. Keshet, "Fooling end-to-end speaker verification with adversarial examples," in *ICASSP*, 2018.
[5] X. Li, J. Zhong, X. Wu, J. Yu, X. Liu, and H. Meng, "Adversarial attacks on gmm i-vector based speaker verification systems," in *ICASSP*, 2020.
[6] A. Jati, C.-C. Hsu, M. Pal, R. Peri, W. AbdAlmageed, and S. Narayanan, "Adversarial attack and defense strategies for deep speaker recognition systems," *Computer Speech & Language*, vol. 68, p. 101199, 2021.
[7] W. Zhang, S. Zhao, L. Liu, J. Li, X. Cheng, T. F. Zheng, and X. Hu, "Attack on practical speaker verification system using universal adversarial perturbations," in *ICASSP*, 2021.
[8] J. Li, X. Zhang, C. Jia, J. Xu, L. Zhang, Y. Wang, S. Ma, and W. Gao, "Universal adversarial perturbations generative network for speaker recognition," in *ICME*, 2020.
[9] Y. Xie, Z. Li, C. Shi, J. Liu, Y. Chen, and B. Yuan, "Enabling fast and universal audio adversarial attack using generative model," in *AAAI*, 2021.
[10] Q. Wang, P. Guo, and L. Xie, "Inaudible adversarial perturbations for targeted attack in speaker recognition," in *INTERSPEECH*, 2020.
[11] A. S. Shamsabadi, F. S. Teixeira, A. Abad, B. Raj, A. Cavallaro, and I. Trancoso, "Foolhd: Fooling speaker identification by highly imperceptible adversarial disturbances," in *ICASSP*, 2021.
[12] G. Chen, S. Chen, L. Fan, X. Du, Z. Zhao, F. Song, and Y. Liu, "Who is real Bob? adversarial attacks on speaker recognition systems," in *S&P*, 2021.
[13] T. Du, S. Ji, J. Li, Q. Gu, T. Wang, and R. Beyah, "Sirenattack: Generating adversarial audio for end-to-end acoustic systems," in *ASIACCS*, 2020.
[14] Y. Gong and C. Poellabauer, "Crafting adversarial examples for speech paralinguistics applications," *CoRR*, vol. abs/1711.03280, 2017.
[15] H. Abdullah, M. S. Rahman, W. Garcia, L. Blue, K. Warren, A. S. Yadav, T. Shrimpton, and P. Traynor, "Hear "no evil", see "kenansville": Efficient and transferable black-box attacks on speech recognition and voice identification systems," in *IEEE S&P*, 2021.
[16] H. Wu, Y. Zhang, Z. Wu, D. Wang, and H. Lee, "Voting for the right answer: Adversarial defense for speaker verification," in *INTERSPEECH*, 2021.
[17] R. Olivier, B. Raj, and M. Shah, "High-frequency adversarial defense for speech and audio," in *ICASSP*, 2021.
[18] F. Tramèr, N. Carlini, W. Brendel, and A. Madry, "On adaptive attacks to adversarial example defenses," in *NeurIPS*, 2020.
[19] A. Athalye, N. Carlini, and D. A. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *ICML*, 2018.
[20] D. Wierstra, T. Schaul, T. Glasmachers, Y. Sun, J. Peters, and J. Schmidhuber, "Natural evolution strategies," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 949–980, 2014.
[21] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok, "Synthesizing robust adversarial examples," in *ICML*, 2018.
[22] I. J. Goodfellow, N. Papernot, and P. D. McDaniel, "cleverhans v0.1: an adversarial machine learning library," *CoRR*, vol. abs/1610.00768, 2016.
[23] M. Nicolae, M. Sinn, T. N. Minh, A. Rawat, M. Wistuba, V. Zantedeschi, I. M. Molloy, and B. Edwards, "Adversarial robustness toolbox v1.0.0," *CoRR*, vol. abs/1807.01069, 2018.
[24] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. Speech Audio Process.*, 2011.
[25] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digit. Signal Process.*, 2000.
[26] S. Becker, M. Ackermann, S. Lapuschkin, K.-R. Müller, and W. Samek, "Interpreting and explaining deep neural networks for classification of audio signals," *CoRR*, vol. abs/1807.03418, 2018.
[27] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, and Z. Zhu, "Deep speaker: an end-to-end neural speaker embedding system," *CoRR*, vol. abs/1705.02304, 2017.
[28] S. J. D. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *ICCV*, 2007.
[29] N. Dehak, R. Dehak, J. R. Glass, D. A. Reynolds, P. Kenny *et al.*, "Cosine similarity scoring without score normalization techniques," in *Odyssey*, 2010.
[30] (2020) The most popular acoustic features. http://speech.ee.ntu.edu.tw/~tlkagk/courses/DLHLP20/ASR%20(v12).pdf.
[31] Z. Yang, B. Li, P. Chen, and D. Song, "Characterizing audio adversarial examples using temporal dependency," in *ICLR*, 2019.
[32] W. Xu, D. Evans, and Y. Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," in *NDSS*, 2018.
[33] S. Rebuffi, S. Gowal, D. A. Calian, F. Stimberg, O. Wiles, and T. Mann, "Data augmentation can improve robustness," in *NeurIPS*, 2021.
[34] D. Prabakaran and R. Shyamala, "A review on performance of voice feature extraction techniques," in *Proceedings of the 3rd International Conference on Computing and Communications Technologies*, 2019.
[35] X. Xiao, S. Zhao, D. H. H. Nguyen, X. Zhong, D. L. Jones, E. S. Chng, and H. Li, "Speech dereverberation for enhancement and recognition using dynamic features constrained deep neural networks and feature adaptation," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, no. 1, p. 4, 2016.
[36] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S. Chang, and T. N. Sainath, "Deep learning for audio signal processing," *IEEE J. Sel. Top. Signal Process.*, vol. 13, no. 2, pp. 206–219, 2019.
[37] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS-W*, 2017.
[38] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *ICLR*, 2015.
[39] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *ICLR*, 2018.
[40] X. Yuan, Y. Chen, Y. Zhao, Y. Long, X. Liu, K. Chen, S. Zhang, H. Huang, X. Wang, and C. A. Gunter, "Commandersong: A systematic approach for practical adversarial voice recognition," in *USENIX Security*, 2018.
[41] H. Kwon, H. Yoon, and K.-W. Park, "Poster: Detecting audio adversarial example through audio modification," in *CCS*, 2019.

[42] K. Rajaratnam, B. Alshemali, and J. Kalita, "Speech coding and audio preprocessing for mitigating and detecting audio adversarial examples on automatic speech recognition," http://cs.uccs.edu/~jkalita/work/reu/REU2018/07Rajaratnam.pdf, 2018.

[43] K. Vos, K. V. Sørensen, S. S. Jensen, and J.-M. Valin, "Voice coding with opus," in *Audio Engineering Society Convention*, 2013.

[44] J. Valin, "Speex: A free codec for free speech," *CoRR*, vol. abs/1602.08668, 2016.

[45] E. Ekudden, R. Hagen, I. Johansson, and J. Svedberg, "The adaptive multi-rate speech coder," in *Workshop on Speech Coding*, 1999.

[46] M. Bosi, K. Brandenburg, S. Quackenbush, L. Fielder, K. Akagiri, H. Fuchs, and M. Dietz, "Iso/iec mpeg-2 advanced audio coding," *Journal of the Audio engineering society*, vol. 45, no. 10, pp. 789–814, 1997.

[47] S. Hacker, *MP3: The definitive guide*. O'Reilly Sebastopol, 2000.

[48] J. Benesty, *Springer handbook of speech processing*, ser. Springer Handbooks, 2008.

[49] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE signal processing letters*, vol. 6, no. 1, pp. 1–3, 1999.

[50] R. Xu and D. C. W. II, "Survey of clustering algorithms," *IEEE Trans. Neural Networks*, 2005.

[51] L. A. Leiva and E. Vidal, "Warped k-means: An algorithm to cluster sequentially-distributed data," *Inf. Sci.*, vol. 237, pp. 196–210, 2013.

[52] "Ivector-plda model released by kaldi," https://kaldi-asr.org/models/m7, 2022.

[53] G. Chen, Z. Zhao, F. Song, S. Chen, L. Fan, and Y. Liu, "AS2T: Arbitrary source-to-target adversarial attack on speaker recognition systems," *IEEE Transactions on Dependable and Secure Computing*, 2022.

[54] Z. Chen, L.-C. Chang, C. Chen, G. Wang, and Z. Bi, "Defending against fakebob adversarial attacks in speaker verification systems with noise-adding," *Algorithms*, 2022.

[55] X. Zhang, X. Zhang, M. Sun, X. Zou, K. Chen, and N. Yu, "Imperceptible black-box waveform-level adversarial attack towards automatic speaker recognition," *Complex & Intelligent Systems*, 2022.

[56] M. Pal, A. Jati, R. Peri, C. Hsu, W. AbdAlmageed, and S. Narayanan, "Adversarial defense for deep speaker recognition using hybrid adversarial training," in *ICASSP*, 2021.

[57] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *ICASSP*, 2015.

[58] N. Carlini and D. A. Wagner, "Towards evaluating the robustness of neural networks," in *S&P*, 2017.

[59] L. Bu, Z. Zhao, Y. Duan, and F. Song, "Taking care of the discretization problem: A comprehensive study of the discretization problem and a black-box adversarial attack in discrete integer domain," *IEEE TDSC*, 2021.

[60] H. Tan, L. Wang, H. Zhang, J. Zhang, M. Shafiq, and Z. Gu, "Adversarial attack and defense strategies of speaker recognition systems: A survey," *Electronics*, vol. 11, 2022.

[61] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs," in *ICASSP*, 2001.

[62] Y. Xiang, G. Hua, and B. Yan, *Digital audio watermarking: fundamentals, techniques and challenges*. Springer, 2017.

[63] H. Abdullah, K. Warren, V. Bindschaedler, N. Papernot, and P. Traynor, "Sok: The faults in our asrs: An overview of attacks against automatic speech recognition and speaker identification systems," in *S&P*, 2021.

[64] D. W. Griffin and J. S. Lim, "Signal estimation from modified short-time fourier transform," in *ICASSP*. IEEE, 1983.

[65] N. Carlini and D. A. Wagner, "Audio adversarial examples: Targeted attacks on speech-to-text," in *SPW*, 2018.

[66] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[67] M. Müller, "Dynamic time warping," *Information retrieval for music and motion*, pp. 69–84, 2007.

[68] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.

[69] Y. Li, L. Li, L. Wang, T. Zhang, and B. Gong, "NATTACK: learning the distributions of adversarial examples for an improved black-box attack on deep neural networks," in *ICML*, 2019.

[70] Y. Dong, H. Su, B. Wu, Z. Li, W. Liu, T. Zhang, and J. Zhu, "Efficient decision-based black-box adversarial attacks on face recognition," in *CVPR*, 2019.

[71] L. Blue, L. Vargas, and P. Traynor, "Hello, is it me you're looking for?: Differentiating between human and electronic speakers for voice interface security," in *WiSec*, 2018.

[72] Y. Meng, J. Li, M. Pillari, A. Deopujari, L. Brennan, H. Shamsie, H. Zhu, and Y. Tian, "Your microphone array retains your identity: A robust voice liveness detection system for smart speaker," in *USENIX Security*, 2022.

[73] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A large-scale speaker identification dataset," in *INTERSPEECH*, 2017.

[74] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *NeurIPS*, 2020.

[75] N. Vaessen and D. A. van Leeuwen, "Fine-tuning wav2vec2 for speaker recognition," in *ICASSP*, 2022.

[76] C. Feng, P. Hsu, and H. Lee, "Silence is sweeter than speech: Self-supervised model using silence to store speaker information," *CoRR*, vol. abs/2205.03759, 2022.

[77] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, "Libritts: A corpus derived from librispeech for text-to-speech," in *Interspeech*, 2019.

[78] M. Sharif, L. Bauer, and M. K. Reiter, "On the suitability of lp-norms for creating and preventing adversarial examples," in *CVPR Workshops*, 2018.

[79] B. Zheng, P. Jiang, Q. Wang, Q. Li, C. Shen, C. Wang, Y. Ge, Q. Teng, and S. Zhang, "Black-box adversarial attacks on commercial speech platforms with minimal information," in *CCS*, 2021.

[80] Z. Zhao, G. Chen, J. Wang, Y. Yang, F. Song, and J. Sun, "Attack as defense: Characterizing adversarial examples using robustness," in *ISSTA*, 2021.

[81] H. Abdullah, W. Garcia, C. Peeters, P. Traynor, K. R. B. Butler, and J. Wilson, "Practical hidden voice attacks against speech and speaker recognition systems," in *NDSS*, 2019.

[82] E. Wenger, M. Bronckers, C. Cianfarani, J. Cryan, A. Sha, H. Zheng, and B. Y. Zhao, ""hello, it's me": Deep learning-based speech synthesis attacks in the real world," in *CCS*, 2021.

[83] Y. Chen, J. Zhang, X. Yuan, S. Zhang, K. Chen, X. Wang, and S. Guo, "Sok: A modularized approach to study the security of automatic speech recognition systems," *CoRR*, vol. abs/2103.10651, 2021.

[84] Y. Qin, N. Carlini, G. W. Cottrell, I. J. Goodfellow, and C. Raffel, "Imperceptible, robust, and targeted adversarial examples for automatic speech recognition," in *ICML*, 2019.

[85] L. Schönherr, K. Kohls, S. Zeiler, T. Holz, and D. Kolossa, "Adversarial attacks against automatic speech recognition systems via psychoacoustic hiding," in *NDSS*, 2019.

[86] Z. Li, Y. Wu, J. Liu, Y. Chen, and B. Yuan, "Advpulse: Universal, synchronization-free, and targeted audio adversarial attacks via subsecond perturbations," in *CCS*, 2020.

[87] R. Taori, A. Kamsetty, B. Chu, and N. Vemuri, "Targeted adversarial examples for black box audio systems," in *SPW*, 2019.

[88] T. Eisenhofer, L. Schönherr, J. Frank, L. Speckemeier, D. Kolossa, and T. Holz, "Dompteur: Taming audio adversarial examples," in *USENIX Security*, 2021.

[89] D. Mukhopadhyay, M. Shirvanian, and N. Saxena, "All your voices are belong to us: Stealing voices to fool humans and machines," in *ESORICS*, 2015.

[90] Y. Xie, Z. Li, C. Shi, J. Liu, Y. Chen, and B. Yuan, "Real-time, robust and adaptive universal adversarial attacks against speaker recognition systems," *J. Signal Process. Syst.*, 2021.

[91] P. Samangouei, M. Kabkab, and R. Chellappa, "Defense-gan: Protecting classifiers against adversarial attacks using generative models," in *ICLR*, 2018.

[92] S. Joshi, J. Villalba, P. Zelasko, L. Moro-Velázquez, and N. Dehak, "Study of pre-processing defenses against adversarial attacks on state-of-the-art speaker recognition systems," *TIFS*, 2021.

[93] X. Li, N. Li, J. Zhong, X. Wu, X. Liu, D. Su, D. Yu, and H. Meng, "Investigating robustness of adversarial samples detection for automatic speaker verification," in *INTERSPEECH*, 2020.

[94] Z. Peng, X. Li, and T. Lee, "Pairing weak with strong: Twin models for defending against adversarial attack on speaker verification," in *INTERSPEECH*, 2021.

[95] N. Carlini and D. A. Wagner, "Adversarial examples are not easily detected: Bypassing ten detection methods," in *AISec@CCS*, 2017.

[96] I. Viñals, A. Ortega, A. Miguel, and E. Lleida, "An analysis of the short utterance problem for speaker characterization," *Applied Sciences*, 2019.
[97] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *MHS*, 1995.
[98] D. Arthur and S. Vassilvitskii, "k-means++: the advantages of careful seeding," in *SODA*, 2007.

Guangke Chen received his BEng degree from South China University of Technology, Guangzhou, China, in 2019. He is currently pursuing the Ph.D. degree with ShanghaiTech University, advised by Dr. Song. His research interest lies in the area of multimedia and machine learning security and privacy. He is currently doing research on the security issues of speaker and speech recognition systems. More information is available at http://guangkechen.site/.

Zhe Zhao received his B.S. degree from Ocean University of China, Tsingtao, China, in 2016. From 2016 to 2018, he was a software engineer at Hewlett-Packard Company. Now he is a Ph.D. student at School of Information Science and Technology, ShanghaiTech University. His research interest lies in the area of software engineering and testing. He is currently doing research in trusted artificial intelligence. His supervisor is Dr. Song.

Fu Song received the B.S. degree from Ningbo University, Ningbo, China, in 2006, the M.S. degree from East China Normal University, Shanghai, China, in 2009, and the Ph.D. degree in computer science from University Paris-Diderot, Paris, France, in 2013. From 2013 to 2016, he was a Lecturer and Associate Research Professor at East China Normal University. From August 2016 to July 2021, he is an Assistant Professor with ShanghaiTech University, Shanghai, China. Since July 2021, he is an Associate Professor with ShanghaiTech University. His research interests include formal methods and computer/AI security.

Sen Chen (Member, IEEE) is an Associate Professor at Tianjin University, China. Before that, he was a Research Assistant Professor at Nanyang Technological University (NTU), Singapore, and a Research Assistant of NTU from 2016 to 2019 and a Research Fellow from 2019-2020. He received his Ph.D. degree in Computer Science East China Normal University, China, in 2019. His research focuses on Security and Software Engineering. More information is available on https://sen-chen.github.io/.

Lingling Fan is an Associate Professor at Nankai University, China. She received her Ph.D and BEng degrees in computer science from East China Normal University, Shanghai, China in June 2019 and June 2014, respectively. In 2017, she joined Nanyang Technological University (NTU), Singapore as a Research Assistant and then had been as a Research Fellow of NTU since 2019. Her research focuses on program analysis and testing, software security. She got an ACM SIGSOFT Distinguished Paper Award at ICSE 2018.

Feng Wang received the bachelor's degree in network engineering from the Nanjing University of Post and Telecommunication, Nanjing, China, in 2016. He obtained master's degree from University of Chinese Academy of Sciences, under the supervision of Prof. Fu Song from ShanghaiTech University in 2019. He is now working at Security Countermeasure Technology Department of Ant Group.

Jiashui Wang is the head of Security Countermeasure Technology Department of Ant Group and the main founder of Ant Security Light-Year Lab.

Supplemental Material of The Article Entitled Towards Understanding and Mitigating Audio Adversarial Examples for Speaker Recognition

Guangke Chen, Zhe Zhao, Fu Song, Sen Chen, Lingling Fan, Feng Wang, and Jiashui Wang

APPENDIX A
SUPPLEMENTAL MATERIAL

A.1 Details of the Datasets

Spk_{10}-enroll consists of 10 speakers (5 males and 5 females), 10 voices per speaker. The speakers are randomly selected from the "test-other" and "dev-other" subsets of the popular dataset Librispeech [6], [7], [8], [12], [57]. For each speaker, we select the top-10 longest voices in order to have better enrollment embedding [96]. The voices in Spk_{10}-enroll are used for speaker enrollment of the CSI-E, SV, and OSI tasks. Spk_{10}-test consists 10 speakers (5 males and 5 females), 100 randomly selected voices per speaker. Spk_{10}-test has the same speakers as Spk_{10}-enroll, but distinct voices.

Both Spk_{251}-train and Spk_{251}-test are taken from the "train-clean-100" subset of Librispeech, each of which has the same 251 speakers (126 males and 125 females). Following [6], for each speaker, 90% of his/her voices are added into Spk_{251}-train, and the remaining 10% are added into Spk_{251}-test. Spk_{251}-train is used to train background models while Spk_{251}-test is used for adversarial attacks on the CSI-NE task. Note that there are no overlapping speakers among Spk_{251}-train, Spk_{10}-enroll and Spk_{10}-test.

A.2 Attacks

A plethora of adversarial attacks have been proposed, most of which are primarily studied in computer vision. It is largely unknown if they can successfully be ported to the speaker recognition domain. Thus, we only consider the attacks that have been demonstrated to be effective on at least one speaker recognition task, including four white-box attacks: Fast Gradient Sign Method (FGSM) [38], Projected Gradient Descent (PGD) [39], Carlini and Wagner's attack (CW) [58], an integration of the CW and PGD attacks, and three black-box attacks: FAKEBOB [12], SirenAttack [13], and Kenansville [15].

FGSM perturbs an input x by performing one-step gradient ascent to maximize a loss function. Formally, a potential adversarial example is:

$$\hat{x} = x + \epsilon \times sign(\nabla_x \mathcal{L}(x, y)),$$

where ϵ is the step_size of gradient ascent, *sign* is the sign function, and $\mathcal{L}(x, y)$ is the loss function describing the cost of classifying x as label y.

PGD is an iterative version of FGSM. In each iteration, PGD applies FGSM with a small step_size α and clips the result to ensure that it stays within an ϵ-neighborhood of the original input x. The intermediate example after the i-th iteration is:

$$x^i = clip_{x,\epsilon}(x^{i-1} + \alpha \times sign(\nabla_x \mathcal{L}(x^{i-1}, y))).$$

Note that the PGD attack starts from a randomly perturbed example, which helps the attack find a better local optimum. We denote by PGD-n the PGD attack with n iteration steps, where the larger n is, the stronger the attack is.

CW is introduced to search for adversarial examples with the small magnitude of perturbations. It formulates finding adversarial examples as an optimization problem with the objective function $\mathcal{L}(x, y) + c \times \|x - x_0\|_p$. The first term measures the effectiveness such that $\mathcal{L}(x, y) \leq 0$ if and only if the attack succeeds. For untargeted (resp. targeted) attack, $\mathcal{L}(x, y) = [S(x)]_y - \max_{i \neq y}[S(x)]_i + \kappa$ (resp. $\max_{i \neq y}[S(x)]_i - [S(x)]_y + \kappa$) where the parameter κ controls the confidence of the adversarial examples and the larger κ is, the stronger the adversarial examples are. The second term quantifies the imperceptibility by the L_p ($p = 0, 1, 2, \infty$) distance between adversarial and original examples, leading to three versions of CW attack, denoted by CW_0, CW_2, and CW_∞, respectively. The parameter c is used as the trade-off between the effectiveness and imperceptibility, and the optimal value of c is found by a binary search. We consider CW_2 and CW_∞ in this work and denote by CW_2-x (resp. CW_∞-n) the CW_2 (resp. CW_∞) attack with $\kappa = x$ (resp. n iteration steps).

- *Guangke Chen is with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China, and with the SKLCS, Institute of Software, Chinese Academy of Sciences, Beijing, China.*
- *Zhe Zhao is with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China.*
- *Fu Song (corresponding author) is with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China. Email: songfu@shanghaitech.edu.cn*
- *Sen Chen is with the College of Intelligence and Computing, Tianjin University, Tianjin, China.*
- *Lingling Fan is with the College of Cyber Science, Nankai University, Tianjin, China.*
- *Feng Wang and Jiashui Wang are with the Ant Group, Zhejiang, China.*

Integration of the CW and PGD attacks (i.e., CW_∞ in our evaluation) uses the loss function of the CW attack but optimized by PGD, the same as [39], to improve the attack efficiency.

FAKEBOB is similar to PGD except that it estimates gradients via Natural Evolution Strategy (NES) [20] that only relies on the output of the model. NES first creates m noisy examples by adding Gaussian noises onto an example. Then, the values of the loss function of m examples are obtained by querying the model, which are finally exploited to approximate the gradient. FAKEBOB adopts an early-stop strategy to reduce the number of queries, i.e., stop searching once an adversarial example is found. Similar to the CW attack, FAKEBOB also provides an option to control the confidence of adversarial examples via a parameter κ. FAKEBOB also proposed the first algorithm to estimate the threshold for SV and OSI tasks. One of the crucial parameter of FAKEBOB is the samples_per_draw m of NES.

SirenAttack is a gradient-free black-box attack based on Particle Swarm Optimization (PSO) [97] that only relies on the output of the model. PSO maintains a swarm of particles, each of which is a candidate solution to the optimization problem. They are iteratively updated via the weighted linear combination of three parts, i.e., inertia, local best solution, and global best solution. When the algorithm terminates, the global best solution returns an optima. SirenAttack runs the PSO subroutine multiple times (each run called an epoch) and globally keeps track of the best solution. The crucial parameter of SirenAttack include the number of epochs $epoch_{max}$, the number of iterations $iter_{max}$ in each epoch, and the number of particles $n\_particles$ used in PSO.

Kenansville is a signal-processing based attack which crafts adversarial voices by decomposing benign voices and then reconstructing voices using part of the decomposing information (other decomposing information is discarded). The amount of information used in reconstruction is controlled by the attack factor which has the max value max_attack_factor and will be iteratively updated via a binary search within $max\_iteration$ to improve the imperceptibility of the attack. Kenansville features two signal processing methods, i.e., Fast Fourier Transform (FFT) and Singular Spectrum Analysis (SSA), where FFT method is not considered in our evaluation since it is much less effective than the SSA method [63].

A.3 Tuning the Parameters of Transformations

To tune the parameters of the transformations, we vary the parameters as shown in Table 8 and conduct all the attacks mentioned in Section 8.

The results are depicted as curves in Fig. 7, Fig. 8, and Fig. 9. We choose the optimal parameters according to the R1 score on FGSM, as R1 score assigns equal importance to the accuracy on benign examples and the accuracy on adversarial examples. We consider FGSM as it is the weakest one among all the attacks, as shown in the (Baseline) row of TABLE 4, and a good parameter should provide strong resilience to the weakest attack. Although these optimal parameters may not be the optimal ones against the other attacks, they are still very promising.

TABLE 8: The ranges and optimal values for parameters of transformations.

Transformation (Parameter)	Range	Optimal
QT (q)	128, 256, 512, 1024	512
AT (snr)	2 to 20 dB, step 2 dB	16 dB
AS (k)	3 to 21, step 2	17
MS (k)	3 to 21, step 2	7
DS (τ)	0.05 to 0.95, step 0.05	0.45
LPF (f_p, f_s)	f_p: 4000 Hz f_s: 4500 to 8000 Hz, step 500 Hz	f_s=4500 Hz
BPF (f_{pl}, f_{pu}, f_{sl}, f_{su})	f_{pl}: 300 Hz f_{pu}: 4000 Hz f_{sl}: 50 to 200 Hz, step 50 Hz f_{su}: 5000 Hz to 8000 Hz, step 500 Hz	f_{sl}=150 Hz f_{su}=6000 Hz
OPUS (b_o)	6-20 kbps, step 1 kbps	8 kbps
SPEEX (b_s)	4-44 kbps, step 2 kbps	11 kbps
AMR (b_r)	6.6, 8.85, 12.65, 14.25, 15.85 18.25, 19.85, 23.05, 23.85 kbps	6.6 kbps
AAC-V (q_c)	1-5, step 1	1
AAC-C (b_c)	15-85 kbps, step 5 kbps	15 kbps
MP3-V (q_m)	0-9, step 1	4
MP3-C (b_m)	8, 16, 24, 32, 40, 48, 64, 80, 96, 112, 128, 160 kbps	24 kbps
FeCo (cl_m, cl_r)	cl_m: kmeans/warped-kmeans cl_r: 0.05 to 0.95, step 0.05	FeCo-o(k): cl_r=0.2 FeCo-o(wk): cl_r=0.35 FeCo-d: cl_r=0.1 FeCo-c: cl_r=0.1 FeCo-f: cl_r=0.1

A.4 More Details of Section 5

In this section, we report more detailed results of the evaluation of transformations against non-adaptive attacks.

A.4.1 Impact of Step_size in PGD Attack

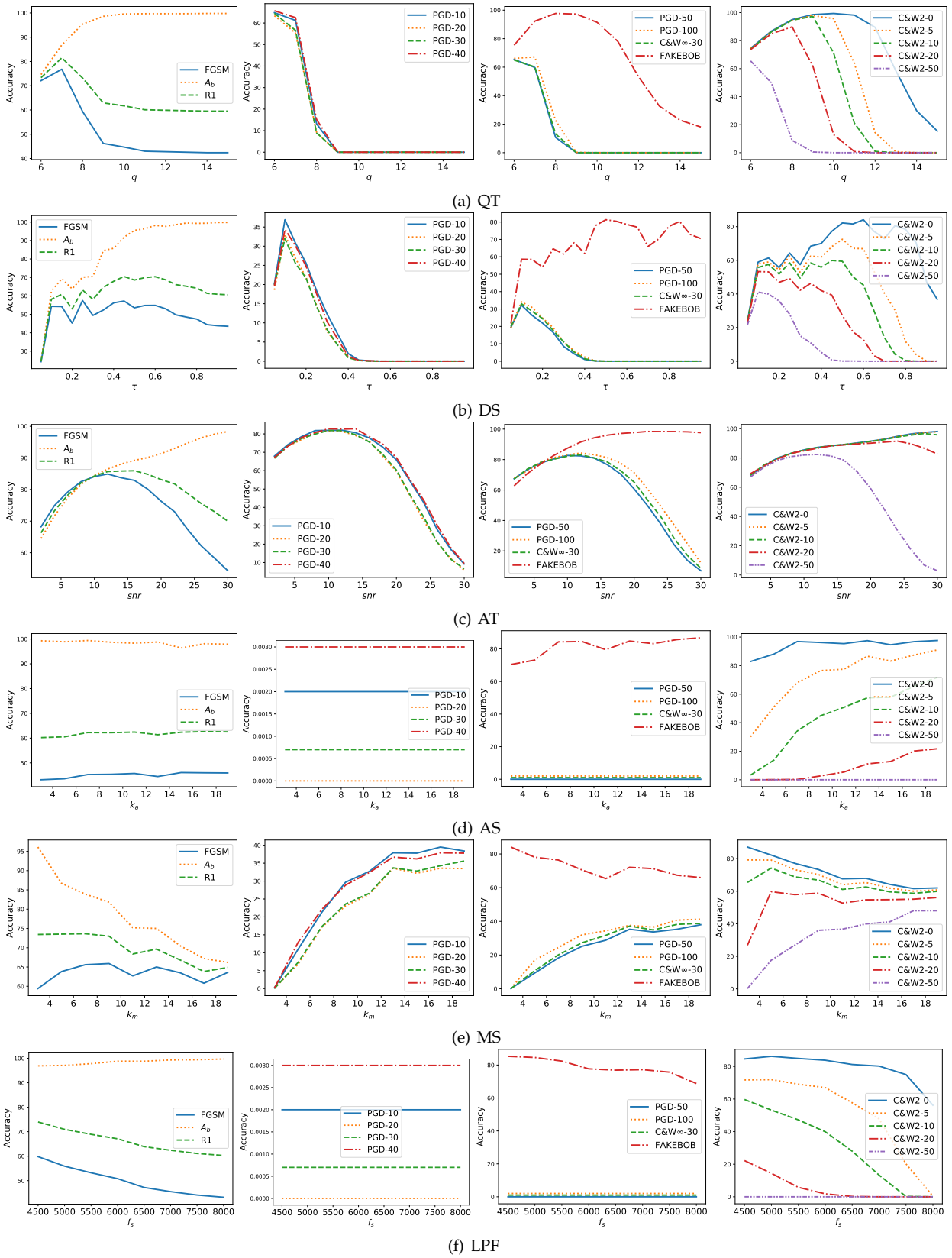
TABLE 9 shows the effectiveness of input transformations in terms of accuracy against the PGD attack when step_size α is fractional to the number #Steps of steps, i.e., $\alpha = \frac{\epsilon}{5\text{\#Steps}}$, $\alpha = \frac{\epsilon}{\text{\#Steps}}$, and $\alpha = \frac{10\times\epsilon}{\text{\#Steps}}$ (Note that previously we set $\alpha = \frac{\epsilon}{5}$). We can observe that increasing the number #Steps of steps and simultaneously decreasing step_size α does not necessarily reduce the effectiveness of input transformations (e.g., QT, AT, MS, OPUS, SPEEX and FeCo-o).

A.4.2 Effectiveness of Transformations w.r.t. Perturbation Budget

Here we conduct experiments to evaluate the effectiveness of transformations on the same attack with different perturbation budgets. We use the white-box attacks FGSM, PGD, and CW_∞ for this experiment since they are stronger than black-box ones. CW2 is excluded since it has no perturbation budget and our existing experiment of varying the confidence value κ of CW_2 already shows that transformations can be defeated by increasing κ at the cost of sacrificing imperceptibility (cf. Findings 3 and Section 8.2). For each attack, we range the perturbation budget from 0.002 to 0.01 with the step of 0.002. The number of steps of PGD and CW_∞ is fixed to 10 and the step size is set to $\frac{\varepsilon}{5}$, the same as before. The results in TABLE 10 show that the adversarial accuracy decreases with the increase of the perturbation budget, at the cost of sacrificing imperceptibility. This is not surprising since the strength of the crafted adversarial voices improves with the increase of ε, as shown in TABLE 11.

A.4.3 More Findings

In this subsection, we discuss in more detail the side effect of transformations on benign examples and usability of transformations.



(a) QT

(b) DS

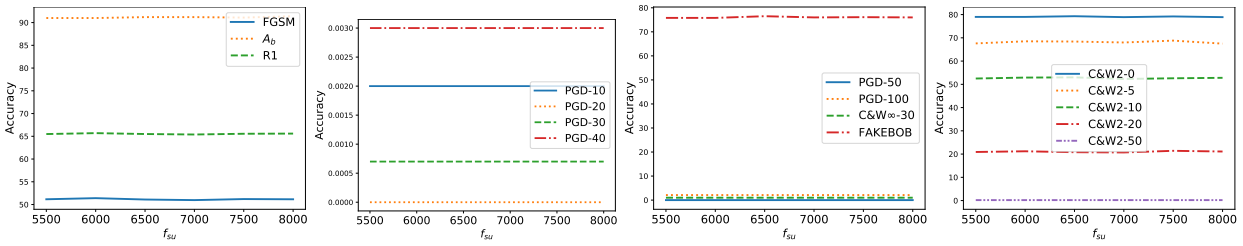
(c) AT

(d) AS

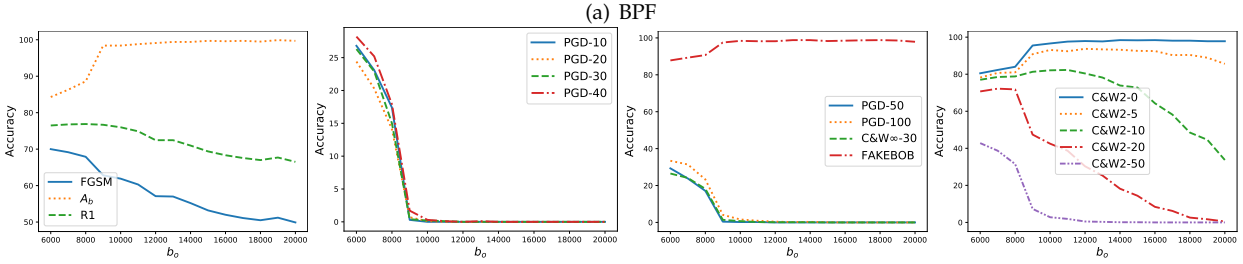
(e) MS

(f) LPF

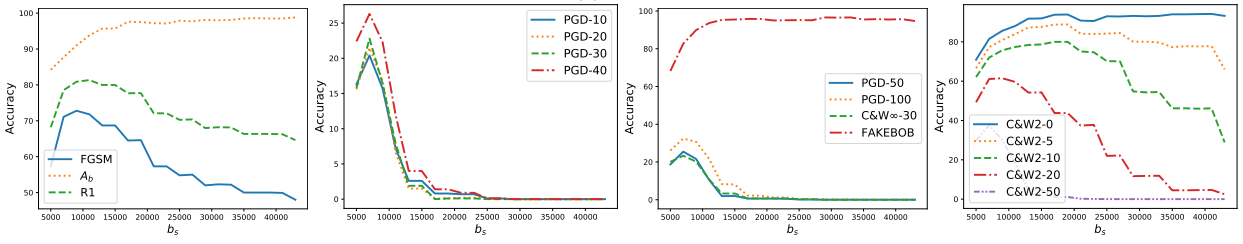
Fig. 7. The performance of input transformations vs. parameter values.



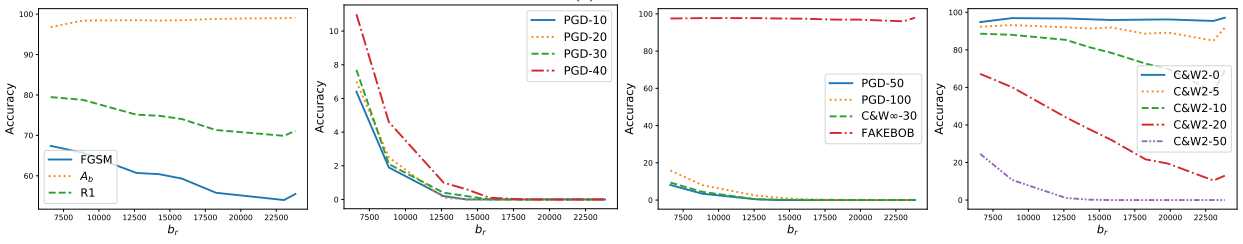
(a) BPF



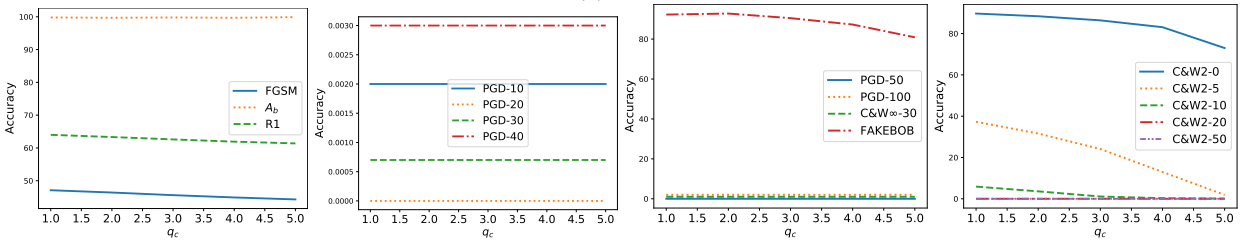
(b) OPUS



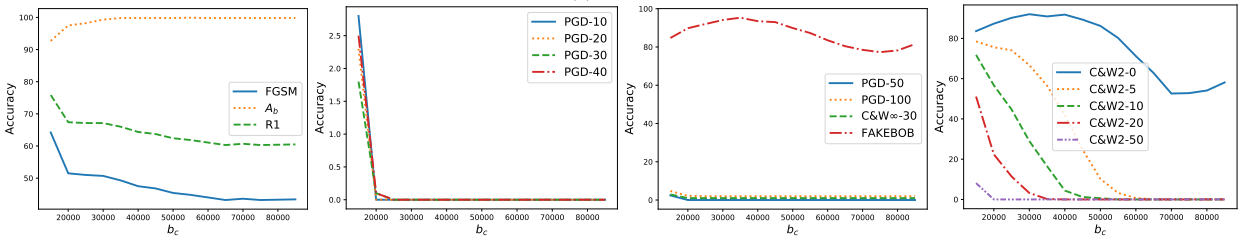
(c) SPEEX



(d) AMR

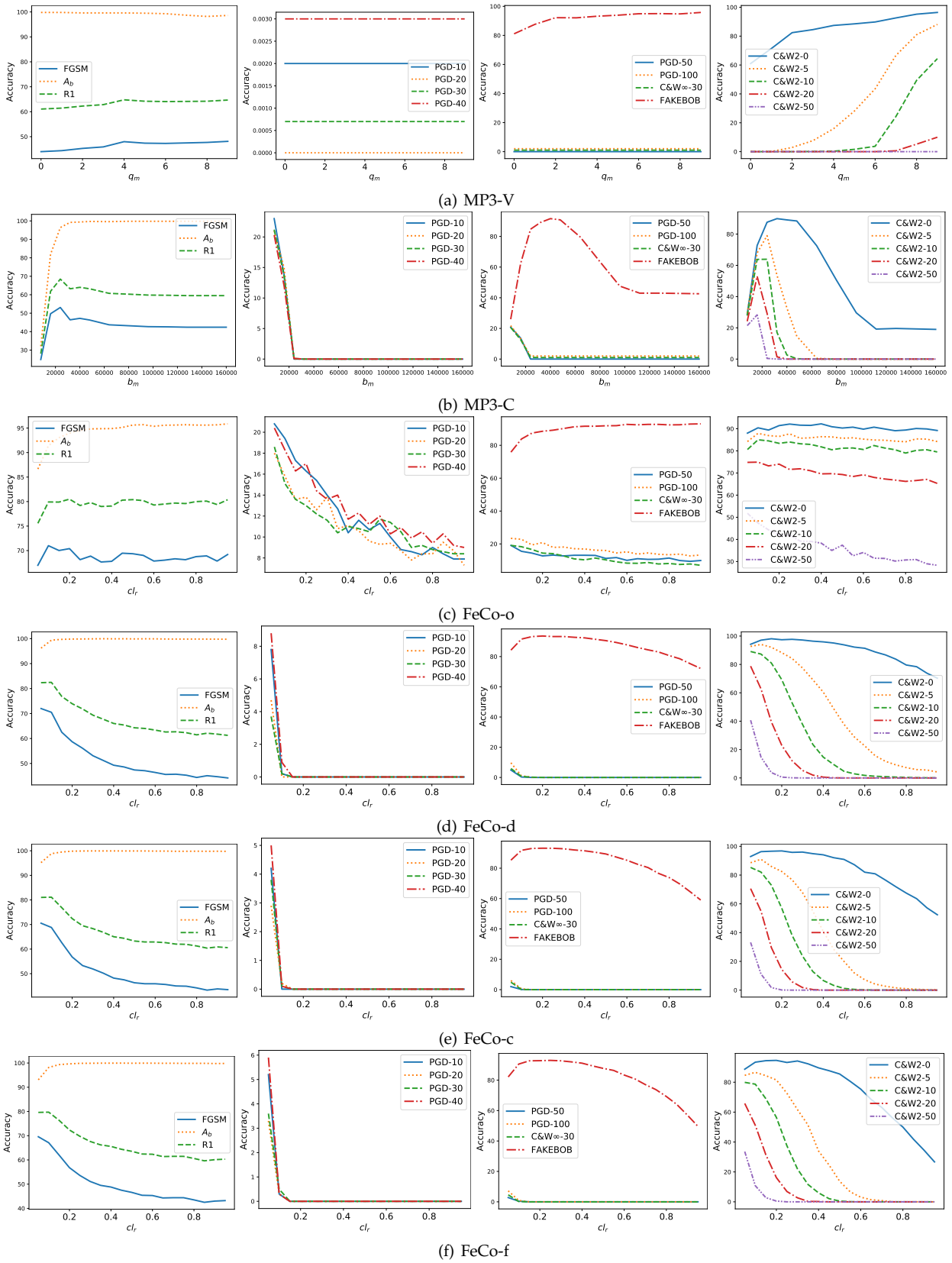


(e) AAC-V



(f) AAC-C

Fig. 8. The performance of input transformations vs. parameter values. For better visualization, we fix $f_{sl} = 150$ Hz of BPF and shows how its performance varies with f_{su}.



(a) MP3-V

(b) MP3-C

(c) FeCo-o

(d) FeCo-d

(e) FeCo-c

(f) FeCo-f

Fig. 9. The performance of input transformations vs. parameter values.

TABLE 9: The effectiveness of input transformations in terms of accuracy (%) against non-adaptive PGD attack when the step_size is fractional to the number of steps (#Steps).

	$\alpha = \frac{\epsilon}{5\text{\#Steps}}$						$\alpha = \frac{\epsilon}{\text{\#Steps}}$						$\alpha = \frac{10\epsilon}{\text{\#Steps}}$					
	10	**20**	**30**	**40**	**50**	**100**	**10**	**20**	**30**	**40**	**50**	**100**	**10**	**20**	**30**	**40**	**50**	**100**
QT	52.4	60.3	64.1	66.3	68.4	71.7	72.1	76.0	77.2	78.4	79.1	81.2	48.7	55.3	58.3	61.7	63.7	70.2
AT	76.3	81.9	85.1	86.8	88.2	90.3	86.4	90.5	91.0	91.8	92.2	92.6	68.6	75.0	78.3	81.0	81.9	87.6
AS	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.3	5.4	9.5	24.8	0.0	0.0	0.0	0.0	0.0	0.0
MS	13.6	22.3	29.1	35.8	38.6	50.4	31.2	39.4	46.6	50.0	52.1	58.0	22.5	17.9	15.1	15.1	14.8	28.8
DS	0.0	0.0	0.1	0.6	1.2	4.1	0.8	0.6	1.5	2.2	3.9	9.5	6.5	0.7	0.0	0.0	0.0	0.1
LPF	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.1	0.3	0.6	1.2	3.8	0.8	0.0	0.0	0.0	0.0	0.1
BPF	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.5	1.2	2.1	3.2	9.1	1.0	0.0	0.0	0.0	0.0	0.0
OPUS	13.7	17.0	21.6	26.2	31.5	44.6	29.9	38.2	44.6	48.2	51.9	59.3	31.8	19.1	15.2	14.3	15.4	26.9
SPEEX	1.7	2.9	6.3	9.9	14.2	28.7	6.4	14.1	21.9	27.0	31.7	44.0	16.8	7.5	4.8	4.5	5.5	14.6
AMR	5.5	8.5	14.9	19.5	24.4	38.4	11.1	21.1	29.3	34.6	39.9	48.7	18.5	10.6	5.8	6.1	7.4	17.8
AAC-V	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.7	0.0	0.0	0.0	0.0	0.0	0.0
AAC-C	1.5	1.7	3.0	4.5	6.0	13.1	2.3	4.0	6.3	7.3	10.1	19.2	14.3	1.5	0.3	0.4	0.4	1.3
MP3-V	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
MP3-C	0.0	0.1	0.1	0.3	0.8	3.1	0.2	0.3	1.7	2.5	3.0	7.8	1.9	0.1	0.0	0.0	0.0	0.0
FeCo-o(k)	10.4	12.9	14.9	19.3	20.5	29.4	22.4	26.5	27.7	33.1	36.3	38.2	40.3	22.1	19.0	16.7	17.3	22.1
FeCo-d(k)	0.6	0.6	1.2	0.9	2.3	5.6	1.2	3.7	5.3	6.2	8.0	13.5	9.2	2.7	1.0	1.0	1.6	4.8
FeCo-c(k)	0.5	0.6	1.2	1.2	1.4	6.6	1.8	2.1	3.9	4.5	6.3	12.2	8.8	1.8	0.8	0.5	1.1	2.9
FeCo-f(k)	0.6	0.7	1.0	1.2	1.7	6.9	1.8	2.2	3.8	4.8	7.0	12.5	8.0	2.0	0.6	0.5	1.5	2.6
FeCo-o(wk)	2.4	2.5	2.8	3.8	3.9	8.2	3.1	3.5	5.7	6.2	8.0	12.6	18.8	9.4	5.7	4.6	4.4	5.0
FeCo-d(wk)	1.9	2.5	2.4	4.1	5.4	12.7	4.3	7.8	11.2	12.5	13.6	21.4	19.2	6.3	3.5	3.0	2.9	7.7
FeCo-c(wk)	1.5	1.6	2.1	2.8	3.7	9.9	2.7	4.8	7.0	8.9	10.5	17.6	16.9	4.8	2.8	2.0	1.8	6.7
FeCo-f(wk)	1.3	1.8	1.9	2.6	3.8	11.2	2.2	5.0	6.8	9.0	10.6	18.1	16.0	5.0	3.2	2.3	2.0	7.5

TABLE 10: Effectiveness of Transformations against non-adaptive attacks w.r.t. the perturbation budget.

Defense	**FGSM**					**PGD**					**CW$_\infty$**				
	0.002	**0.004**	**0.006**	**0.008**	**0.01**	**0.002**	**0.004**	**0.006**	**0.008**	**0.01**	**0.002**	**0.004**	**0.006**	**0.008**	**0.01**
Baseline	42.3	37.2	35.5	34.3	32.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
QT	76.8	61.8	50.6	38.8	38.4	61.2	25.7	12.4	9.4	9.4	60.7	25.8	12.9	9.6	9.6
AT	82.9	63.6	49.7	43.3	36.9	77.8	37.2	12.4	3.3	1.8	77.9	37.1	14.6	4.8	3.0
AS	46.0	41.9	39.1	38.2	35.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
MS	65.6	57.6	52.4	48.1	44.5	21.3	12.0	9.4	8.4	8.3	21.2	11.7	9.7	8.8	8.4
DS	57.2	50.4	47.3	46.1	44.8	0.3	0.3	0.3	0.3	0.3	0.2	0.2	0.2	0.2	0.2
LPF	59.8	56.1	52.1	52.4	49.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
BPF	51.4	45.7	41.6	40.0	39.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
OPUS	67.9	61.8	58.7	56.6	55.6	17.4	7.6	4.3	3.7	3.3	17.0	7.2	5.0	3.8	3.5
SPEEX	71.8	63.0	58.8	53.1	48.5	7.2	5.7	5.6	5.6	5.6	6.7	5.5	5.4	5.4	5.4
AMR	67.4	59.8	55.8	51.0	48.6	6.4	1.3	0.9	0.9	0.9	5.8	1.7	1.4	1.3	1.3
AAC-V	47.1	40.6	38.6	37.4	35.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
AAC-C	64.2	54.2	48.7	47.5	43.6	2.8	1.7	1.7	1.7	1.7	3.2	2.3	2.2	2.2	2.2
MP3-V	48.0	42.2	39.6	37.6	36.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
MP3-C	53.1	44.2	39.0	37.4	35.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
FeCo-o(k)	70.4	56.8	46.1	36.6	32.4	16.3	7.1	4.9	4.4	4.3	14.1	5.1	2.7	1.9	1.5

TABLE 11: The imperceptibility and loss of non-adapative attacks w.r.t. the perturbation budget.

Attack	ϵ	**Imperceptibility**		**Loss**	
		SNR	**PESQ**	$\mathbf{L}_{CE}$	$\mathbf{L}_{M}$
FGSM	**0.002**	28.92	2.24	-3.99	-2.28
	0.004	22.86	1.66	-4.19	-2.89
	0.006	19.37	1.41	-4.24	-3.15
	0.008	16.85	1.28	-4.23	-3.29
	0.01	14.93	1.20	-4.28	-3.44
PGD	**0.002**	33.22	2.88	-45.83	-45.82
	0.004	27.54	2.21	-49.48	-49.47
	0.006	24.11	1.87	-50.95	-50.94
	0.008	21.71	1.65	-51.50	-51.50
	0.01	20.41	1.56	-51.18	-51.18
CW$_\infty$	**0.002**	33.23	2.88	-44.42	-44.39
	0.004	27.53	2.21	-47.86	-47.83
	0.006	24.11	1.86	-49.48	-49.46
	0.008	21.72	1.65	-50.64	-50.63
	0.01	19.79	1.51	-51.36	-51.35

Side effect on benign examples. Most transformations slightly degrade accuracy on benign examples, but the degradation varies. The accuracy degradation reflects the degree of distortions induced by each transformation, i.e., how well the transformation preserves the speech quality. Among all the transformations, QT, AT, MS, and OPUS cause the greatest accuracy degradation ($> 10\%$), indicating that they add more distortions. AAC-V, MP3-V and FeCo-d(k) almost have no side effects, reducing only 0%, 0.2% and 0.4% accuracy on benign examples, respectively. We observe that dynamic bit rate based speech compressions have fewer side effects (e.g., MP3-V vs. MP3-C, and AAC-V vs. AAC-C), as they preserve the better quality of voices. Among the feature-level transformations, we observe that FeCo-d outperforms the others, indicating that FeCo has fewer effects on the delta features than the others.

Findings 11. Most of transformations can be freely composed with pre-trained models to defeat adversarial examples with slight accuracy degradation on the benign examples. Input transformations with dynamic bit rate and delta feature transformation have the least side effects, while the input transformations QT, AT, MS, and OPUS have the greatest side effects.

Usability of transformations. Though effective transformations against adversarial examples degrade accuracy on benign examples, compared to Baseline, all transformations show good usability in terms of the R1 score. The best one (i.e., AT) improves the R1 score by 77.3% and the worst one (i.e., MP3-V) improves it by 19.4%. This is because in general, the accuracy improvements on adversarial examples are often larger than the accuracy degradation on benign

TABLE 12: Non-adaptive and adaptive SirenAttack against AS, DS, LPF, and BPF when $\epsilon = 0.02$ in terms of model accuracy.

	AS	DS	LPF	BPF
Non-adaptive	42.9%	53.3%	58.0%	48.0%
Adaptive	0%	0%	42.0%	16.3%

examples.

Among the feature-level transformations, we can observe that FeCo-o and FeCo-d often significantly outperform. This is because transformation on preceding features also affects succeeding features, which amplifies the effect of the transformation. Between two clustering algorithms kmeans and warped-kmeans, the effectiveness varies with attacks and in general they are almost comparable. In terms of the R1 score, FeCo-o with kmeans, i.e., FeCo-o(k), ranks the first place.

Findings 12. All transformations exhibit good usability since they lead to significantly better R1 scores. While the transformations QT, AT, and FeCo-o(k) degrade the accuracy on benign examples, they are the three most effective transformations against non-adaptive attacks.

A.5 Approximation of Non-differentiable Transformations by the Identity Function

To measure how accurate it is to substitute a non-differentiable transformation with the identity function, we compute the average L_2 distance between the original voices and the voices after the transformation. The results are shown in Fig. 10, where the L_2 distance is given in the caption of each sub-figure, and the curves in each sub-figure are the waveform of a random chosen voice and the voice after transformation.

From Fig. 10, we can observe that the L_2 distance of QT, AAC-V and MP3-V is much smaller than that of OPUS, SPEEX, AMR, AAC-C and MP3-C, indicating that QT, AAC-V and MP3-V are much closer to the identity function. We can also observe that the difference between the original voice and the voice after transformation of CBR speech compressions is more significant than that of QT and VBR speech compressions (i.e., AAC-V and MP3-V). In conclusion, it seems that it suffices to replace QT and VBR speech compressions with the identity function in the backward pass, but more accurate approximation functions or more advanced adaptive attacks than BPDA are required to circumvent other speech compressions.

A.6 SirenAttack to AS, DS, LPF and BPF

The results are shown in TABLE 12 when $\epsilon = 0.02$ for the adaptive SirenAttack. We can observe that the adaptive SirenAttack reduces the accuracy of these input transformations by at least 16% compared to the non-adaptive one.

A.7 Brute-force Replicate Attack

Replicate attack is not strong due to the randomness of FeCo. The adversary may attempt to improve the attack by enumerating the randomness in a brute-force way. Below we analyze the success probability such a brute-force adversary can achieve.

Suppose the randomness space is $\mathcal{B} = \{B_1, \cdots, B_Q\}$ where B_i denotes a possible clustering result. In each trail of the brute-force, suppose the adversary samples B_a and the victim model samples B_r, we have: $Pr[\text{the attack succeeds}] \geq Pr[B_r = B_a] = \frac{1}{Q}$.

The size of the randomness space Q depends on the duration of the voice and the initial method of the clustering algorithm. For a voice with duration of one second (the minimal duration of the voices in Spk10_test and Spk251_test), the number N of frames is nearly 100. If the initial method is kmeans++ [98], $Q = kN = 500$ ($k = \frac{1}{cl_r} = \frac{1}{0.2} = 5$ for FeCo-o(k)) and $Pr[\text{the attack succeeds}] \geq 0.2\%$. If the initial method is random, $Q = C_{kN}^{N} > 2.04 \times 10^{107}$ and $Pr[\text{the attack succeeds}]$ is close to 0% in the worst case. The success probability of the brute-force attack is very low.

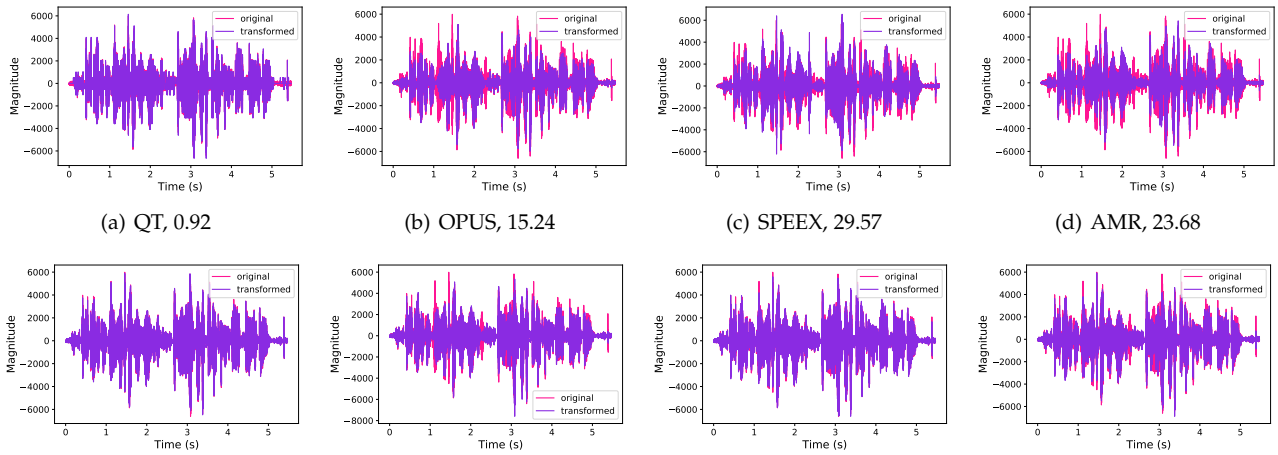
A.8 Defending against Hidden Voice and Speech Synthesis Attacks

To be comprehensive, apart from adversarial attacks, we also evaluate the input transformation-based defenses against hidden voice and speech synthesis attacks under both the non-adaptive and adaptive settings.

Hidden voice and speech synthesis attacks have different attack purposes from adversarial attacks. Given a voice uttered by a source speaker, an adversarial attack intends to perturb the voice such that the perturbed voice is recognized as another speaker by the target SRS, but still recognized as the source speaker by human. In contrast, hidden voice attack aims to craft a perturbed voice which is treated as mere noise by human, but still correctly recognized as the source speaker by the target SRS, and a speech synthesis attack attempts to produce a voice that contains the desired speech content and sound as spoken by the source speaker from the perspective of both human and the target SRS.

For hidden voice attack, we consider the signal processing-based attack in [81]. It generates incomprehensible voices for human by inverting the speech in the time domain (Time Domain Inversion, TDI), accelerating the speed of the speech (Time Scaling, TS), adding high-frequency signal, or generating random phases. We exploit TDI and TS to perturb each speech since they are the two most effective methods [81]. TDI and TS feature the parameters window size w and scaling factor β, respectively, where the smaller w (resp. larger β), the less comprehensible the voices for human and the harder the voices to be correctly recognized by the target SRS. As suggested in [81], we use a linear search to find the optimal parameters where the attack can produce the least understandable voices for human when ensuring the correct recognition of the target SRS. Specifically, to find the optimal w, the TDI attack starts from $w = 1$ milliseconds (ms), gradually increases to 10 ms with step of 0.5 ms, and terminates once the target SRS correctly recognizes the perturbed voice. To find the optimal β, the TS attack starts from $\beta = 20$, gradually reduces to 1.5 with step of 0.5, and terminates once the target SRS correctly recognizes the perturbed voice. Since both TDI and TS are black-box attacks, their adaptive versions are similar to the non-adaptive versions except that the attack terminates once the *defended* SRS correctly recognizes the perturbed voice.

For speech synthesis attack, we exploit the deep learning-based speech synthesis tool used in [82]. The tool



(a) QT, 0.92 (b) OPUS, 15.24 (c) SPEEX, 29.57 (d) AMR, 23.68

(e) AAC-V, 3.87 (f) AAC-C, 6.92 (g) MP3-V, 4.19 (h) MP3-C, 5.92

Fig. 10. The visualization of an original voice and transformed voice by different input transformations. The average L_2 distance between original and transformed voices is listed right of the transformation name.

TABLE 13: Results of input transformations against hidden voice and speech synthesis attacks under both non-adaptive and adaptive settings in terms of attack success rate (%).

	Non-adaptive			Adaptive		
	Hidden		Speech	Hidden		Speech
	TDI	TS	Synthesis	TDI	TS	Synthesis
Baseline	96.1	96.0	96.0	96.1	96.0	96.0
QT	49.2	**43.3**	76.2	77.7	73.9	85.7
AT	55.6	52.7	**72.1**	79.7	**71.8**	**79.1**
AS	**70.0**	**73.7**	**91.0**	**93.8**	89.5	92.5
MS	52.8	49.3	**66.3**	81.2	84.0	84.5
DS	51.2	54.0	**72.4**	79.7	85.3	80.7
LPF	42.4	59.6	88.6	78.5	86.9	**92.8**
BPF	**31.6**	49.0	80.0	70.8	85.1	92.6
OPUS	38.0	49.8	78.3	69.4	79.2	**72.2**
SPEEX	64.0	59.0	80.2	93.2	89.6	**70.6**
AMR	49.5	62.2	88.6	92.7	**93.0**	88.2
AAC-V	**89.9**	**90.3**	**94.9**	**96.5**	**95.2**	**97.9**
AAC-C	**31.6**	53.8	86.9	69.1	88.8	90.2
MP3-V	**88.9**	**90.7**	**95.2**	**95.5**	**96.4**	**98.2**
MP3-C	**23.3**	56.3	88.5	**68.0**	88.4	90.9
FeCo-o(k)	43.3	57.1	83.8	**53.8**	**67.4**	**79.1**
FeCo-d(k)	52.1	49.2	88.2	72.4	74.8	83.2
FeCo-c(k)	52.3	47.8	86.7	**68.8**	75.3	81.0
FeCo-f(k)	52.2	47.6	86.5	68.2	75.0	80.8
FeCo-o(wk)	59.6	60.0	84.5	74.1	75.5	89.5
FeCo-d(wk)	49.8	46.5	87.3	71.8	74.9	87.9
FeCo-c(wk)	52.8	**44.3**	86.3	70.0	**72.5**	86.6
FeCo-f(wk)	52.4	**44.2**	86.8	70.0	72.3	86.6

takes as input a set of voice samples of the source speaker and the desired speech content. We use the voices in Spk10_enroll as the set of voice samples (10 speakers and 10 voices per speaker) and the ten sentences used in [82] as the desired speech content. We consider the following adaptive adversary: (1) running the non-adaptive speech synthesis attack with input voice samples $x_1, \cdots, x_N$. Suppose the output voice is $\hat{x}$; (2) generating adversarial perturbation δ for $\hat{x}$ with the objective of minimizing $\frac{1}{N}\sum_{i=1}^{N} d(Enc(g(\hat{x}+\delta)) - Enc(x_i))$ where g is the input transformation, $Enc(x)$ extracts the embedding (i.e., the vector representing the speaker characteristic) of the voice x, and d measures the distance between two embeddings; (3) using $\hat{x}+\delta$ to attack the *defended* SRS. Specifically, in step (2), we use cosine distance to measure distance between two embeddings and exploit PGD with $\epsilon = 0.002$ and #Steps=50 to craft δ.

The results are shown in TABLE 13. We can observe that all the input transformations are able to reduce the attack success rate under the non-adaptive setting compared to Baseline without any defense, indicating they can also be exploited to mitigate hidden voice and speech synthesis attacks. We also notice that regardless of the attack types, AS, MP3-V and AAC-V are the least effective ones while CBR speech compressions are more effective than VBR ones. However, we should point out some transformations performing differently between adversarial, hidden voice and speech synthesis attacks. While the time-domain W-transformations, especially QT and AT, are quite effective against adversarial attacks, they are not as much effective against hidden voice attack. Input transformations are in general less effective against speech synthesis attack than the other two attacks. The reason is that speech synthesis attack attempts to synthesize high-quality and natural speeches to deceive both human and the target SRS, unlike adversarial and hidden command attacks which perturb the original voice to cause inconsistent recognition between human and the target SRS.

Under the adaptive setting, the adaptive hidden voice attack achieves much higher attack success rate against all the input transformations than the non-adaptive one, e.g., the success rate improves by over 43% on AMR. However, the adaptive speech synthesis attack does not perform better than the non-adaptive one on some input transformations, e.g., OPUS, SPEEX, AMR, and some FeCo. The reason is that the adaptive speech synthesis attack involves solving an optimization problem, and these transformations introduce optimization obstacles, i.e., non-differenetiability of OPUS, SPEEX, and AMR, and the randomness of FeCo.

Findings 13. Input transformations exhibits general defense capability against adversarial, hidden voice, and speech synthesis attacks, although with some differences. Speech synthesis attack is more difficult to defeat by input transformations than the other two attacks.

A.9 Study with Other Models and Datasets

To confirm whether our findings can extend to other speaker recognition models and datasets, we adopt the DeepSpeaker [27] model and the VoxCeleb [73] dataset. Specifically, we randomly select ten speakers (five male and five female) in VoxCeleb and randomly select 110 voices per

TABLE 14: Results of transformations against non-adaptive attacks on DeepSpeaker with VoxCeleb.

Defense	R_1 Score	A_b	A_a												
			L_∞ white-box attacks							L_2 white-box attacks			black-box attacks		
			FGSM	PGD			CW_∞			CW_2			Score-based (L_∞)		Decision-only
				10	20	100	10	20	100	0	0.2	0.5	FAKEBOB	SirenAttack	Kenansville
Baseline	15.6	99.7	48.4	0.4	0.1	0	0	0	0	3.4	0	0	6.9	28.4	22.2
QT	73.0	82.7	70.2	64.1	63.2	67.4	56.8	54.9	59.1	82.5	80.7	66.3	80.9	59.2	43.5
AT	76.2	87.4	71.4	64.2	63	63.9	62.2	60.5	60.8	86.8	86.3	80.4	76.4	55.7	46.7
AS	43.3	94.2	47.9	3.7	3.2	3.6	2.3	1.4	1.3	80.4	63.8	22.6	61.4	50.8	23.5
MS	50.4	96	58.2	7.8	7.2	8.4	5.8	4.1	4.2	87.6	70.7	31.2	69.2	53.3	37
DS	43.5	98.8	54.4	4.5	3.9	3.8	3.1	2.1	1.5	84.4	40.5	0.5	68.5	55.1	40.7
LPF	40.8	99.2	54.3	3.6	3.1	3	2.5	1.7	1.6	86.5	24.3	0	64.5	57.3	31
BPF	38.3	96.2	41.7	4.4	4.1	4	3	2.9	2.8	75.4	41.3	5.3	54.8	44	26.8
OPUS	75.0	98.6	70.3	44.1	44.9	52.7	33	31.4	40.2	96.9	94.4	76.5	83.5	67.5	51.7
SPEEX	76.1	90	73.1	55	56.9	61	53.7	52.5	59.6	89.3	88.1	75.4	80.2	66.2	46.7
AMR	80.5	95.4	77.7	58.5	59	65.5	50	49.9	58.4	94.6	92.5	84.6	88.5	73.3	53.2
AAC-V	39.2	99.5	55.7	3.2	2.9	3.6	1.3	1	1.2	91.5	30.6	0.5	52.1	46.6	27.4
AAC-C	63.2	95.2	72	25.4	24	27.3	18.2	15.4	18.1	90.7	86.6	58	84.8	64.5	30.3
MP3-V	47.3	99.1	60.5	4.4	3.7	4.3	3.6	3.6	3.4	93	50.1	3	80.1	61.5	32.7
MP3-C	52.8	97	58.1	10.3	9.7	11.6	6.7	5.6	5.8	90	74.4	29.3	78.1	54.7	37.7
FeCo-o(wk)-ts	78.8	95.4	72.4	59.1	60.7	65.5	58.8	58.4	63.6	93.7	91.1	81.1	84.6	50.5	33.9
FeCo-o(wk)-rd	70.7	99.1	73.7	32.3	34.7	46.3	21.1	22.4	32	97.2	90.9	66.5	90.1	74.2	32.3

Note: ts and rd denotes different initialization methods [51] of warped-kmeans. The top-3 highest/lowest results are highlighted in blue/red color except for Baseline where no defense is deployed. The accuracy A_a used for computing R1 Score is the average of all the attacks in the same row.

TABLE 15: Imperceptibility and strength of non-adaptive attacks on DeepSpeaker with VoxCeleb.

Attack		Imperceptibility		Loss	
		SNR	PESQ	L_{CE}	L_M
FGSM		25.21	2.16	-2.10	-0.02
PGD-x	**10**	29.84	2.91	-2.71	-0.74
	20	28.90	2.77	-2.81	-0.84
	100	28.47	2.72	-2.93	-0.94
CW_∞-x	**10**	29.99	2.93	-2.47	-0.73
	20	28.99	2.78	-2.53	-0.81
	100	28.50	2.73	-2.62	-0.90
CW_2-κ	**0**	50.44	4.58	-1.95	-0.03
	0.2	44.63	4.41	-2.08	-0.22
	0.5	36.11	3.66	-2.32	-0.52
FAKEBOB		28.18	2.64	-2.01	-0.04
SirenAttack		14.54	1.29	-2.06	-0.01
Kenansville		12.72	1.53	-2.17	-0.02

speaker. The ten longest voices of each speaker are used for enrollment, while the rest 1000 voices (100 voices per speaker $\times$ 10 speakers) in total are used as the benign voices. Unlike ivector-PLDA which models the frame-wise acoustic features, DeepSpeaker also captures the dependency between different frames, so we choose warped-kmeans as the clustering algorithm for FeCo in order to preserve the temporal dependency. In addition, since DeepSpeaker directly accepts as input the original acoustic feature without adding time-derivative features, applying CMVN, or voice activity detection, we only consider the original feature as the transformation point for FeCo. We re-perform experiments of evaluating transformations against non-adaptive attack in Section 5. For PGD and CW_∞, we set the number of steps to 10, 20, and 100, which is a subset of previous number of steps, but are considerably sufficient for our purpose. For CW_2 attack, since DeepSpeaker adopts cosine similarity as the scoring method which has much lower range and scale than PLDA in ivector-PLDA, we set the $\kappa = 0, 0.2, 0.5$. Since the attack success rate of SireAttack attack is too low when ε=0.002, we set ε=0.008 for this attack. The results are shown in TABLE 14 and TABLE 15. All the seven findings in Section 5 still hold with slight difference. Below we analyze Findings 1-Findings 5 and Findings 11-12 one by one.

Findings 1. Time-domain (resp. feature-level) transformations are often more effective than others on L_∞ (resp. L_2) attacks.

For L_∞ attacks (i.e., FGSM, PGD, CW_∞, FAKEBOB, and SirenAttack), the top-3 most effective transformations include two time-domain transformations, while for L_2 attacks (i.e., CW_2), the top-3 most effective transformations include two feature-level transformations.

Findings 2. The effectiveness of input transformations does not necessarily decrease with increase of distortion, since large distortion does not imply stronger adversarial voices.

FGSM, FAKEBOB, SirenAttack, and Kenansville introduce larger distortion than PGD, CW_∞ and CW_2-0.5, but almost all the transformations achieve higher accuracy on FGSM, FAKEBOB, SirenAttack, and Kenansville than PGD, CW_∞ and CW_2-0.5.

Findings 3. The effectiveness of input transformations does not necessarily decrease with increase of attack strength.

The strengths of PGD and CW_∞ improve with the increase of the number of steps, but almost all transformations achieve higher accuracy on PGD-100 (resp. CW_∞-100) than on PGD-10 (resp. CW_∞-10).

Findings 4. AS, LPF, AAC-V, and MP3-V are completely ineffective against attacks that craft high-confidence adversarial voices in non-adaptive setting.

In most cases, AS, LPF, AAC-V, and MP3-V are among the top-3 least effective transformations against high-confidence attacks (PGD, CW_∞, and CW_2-0.5). The exception is that they are not completely ineffective but still provides a little accuracy improvement.

Findings 5. VBR speech compression has less side-effect, but are less effective in mitigating adversarial voices.

AAC-V and MP3-V have higher accuracy on normal examples and lower accuracy on almost adversarial attacks than AAC-C and MP3-C, respectively.

Findings 11. Most of transformations can be freely composed with pre-trained models to defeat adversarial examples with slight accuracy degradation on the benign examples. Input transformations with dynamic bit rate and delta feature transformation have the least side effects, while the input transformations QT, AT, MS, and OPUS have the greatest side effects.

All the transformations except for QT and AT degrades the normal accuracy by no more than 7%. The slight differ-

ences are that DeepSpeaker does not have delta feature and MS and OPUS preserve the normal accuracy well.

Findings 12. All transformations exhibit good usability since they lead to significantly better R1 scores. While the transformations QT, AT, and FeCo-o(k) degrade the accuracy on benign examples, they are the three most effective transformations against non-adaptive attacks.

All the transformations achieve much higher R1 score than the baseline model without any defense. The exception is that the three most effective transformations against non-adaptive attacks are AMR, AT, and FeCo-o(wk)-ts. We remark that QT is still promising.