2026 56th Annual IEEE International Conference on Dependable Systems and Networks (DSN) | 979-8-3315-5149-0/26/$31.00 ©2026 IEEE | DOI: 10.1109/DSN69566.2026.00037

SpeechShield: Latency-Efficient and Robust Timbre-Aware Voice Protection Against Speech Synthesis Deepfake Attacks

Jianshuo Liu[1†], Shiquan Dong[23†], Hong Li[23], Chenghua Gao[23], Kang G. Shin[1]
Haining Wang[4], Yimo Ren[23], Limin Sun[23]
[1]*Department of Computer Science and Engineering, University of Michigan*
[2]*Institute of Information Engineering, Chinese Academy of Sciences*
[3]*School of Cyber Security, University of Chinese Academy of Sciences*
[4]*Department of Electrical and Computer Engineering, Virginia Tech*
{jianshuo,kgshin}@umich.edu,{hnw}@vt.edu,{dongshiquan,lihong,gaochenghua,renyimo,sunlimin}@iie.ac.cn

***Abstract*—Speech synthesis models can generate highly realistic cloned voices by capturing the phonetic features of a speaker. However, adversaries may exploit this capability by collecting users' speech samples (e.g., from social media platforms) without their consent to conduct voice cloning or deepfake attacks. While audio adversarial examples have been proposed as a defense mechanism, most of them rely on time-intensive optimization from scratch in the input space, rendering them impractical for latency-sensitive applications such as live audio streaming. They are also vulnerable to countermeasures such as denoising. To address these weaknesses, we propose SpeechShield, a universal and robust perturbation-generation framework that can work offline or in real time. SpeechShield injects adversarial perturbations into speech features that are critical to timbre, such as harmonic structure and energy spectral patterns. It prevents speech synthesis models from learning accurate voice representations and elevates resistance to adversarial removal. Moreover, SpeechShield projects input audio into a compact high-dimensional latent space to generate meta-perturbations, allowing for efficient fine-tuning adaptation to new samples with minimal delay. By incorporating perceptual objectives, SpeechShield also maintains the intelligibility and naturalness of protected speech. We have conducted extensive experiments on a range of advanced models, datasets, and real-world scenarios, showing that SpeechShield consistently outperforms state-of-the-art (SOTA) defenses in terms of protection effectiveness, transferability, and robustness, all at the lowest latency.**

I. Introduction

With the rapid advancement of generative artificial intelligence (GAI) in the audio domain, models such as ChatGPT and Qwen2-Audio have demonstrated the capability to synthesize highly realistic human speech and engage in naturalistic dialogues. Individuals can now leverage their own voice samples to fine-tune multimodal models such as StyleTTS [1], enabling the generation of synthetic speech that closely resembles their vocal characteristics. However, the abuse/misuse of such models has also posed serious security risks. For instance, adversaries may exploit cloned deepfake voices to impersonate individuals and commit fraud or pass through voice authentication systems, as shown in Fig. 1. Adversaries can scrape publicly available content from victims' social media platforms at no/low cost, and generate cloned voices. Even if victims attempt to limit their online footprint, their voice may still be captured during live-streamed events, where real-time speech capture makes retroactive protection impossible. This calls for the urgent need of reliable technical safeguards against the emerging threats posed by voice-cloning deepfake attacks.

Prior research on defenses against voice-cloning deepfakes has focused on two main directions: (1) adversarial voice generation [2], [3] and (2) voice watermarking [4], [5], [6]. The former perturbs voice samples to degrade speech-synthesis models trained with protected data. In contrast, watermarking embeds imperceptible perturbation patterns into speech, designed to persist in synthesized audio for traceability. However, state-of-the-art (SOTA) watermarking fails to block cloning and is vulnerable to replay attacks. Similarly, adversarial defenses usually search for perturbations in the waveform domain, requiring optimizations on large-scale sample waveform points from scratch, which is computationally ineffective. Moreover, due to the inherently limited training data from the victim, the sequential inference across timestamps (e.g., LSTM [7]) and the narrow receptive fields (e.g., U-Net [8]), existing regression-based models are prone to poor convergence, or simply overfitting, and hence ill-suited for perturbation generation. Adversaries can also readily counter such perturbations with denoising or adversarial training [2], [3], [9]. Thus, it is still challenging to design an effective, efficient anti-voice-cloning framework.

To address these challenges, we propose SpeechShield, a latency-efficient and timbre-aware voice protection framework that is robust against adversarial training and erasing. Unlike existing unstructured perturbations, SpeechShield introduces a novel optimization mechanism that injects energy into acoustic features such as pitch, harmonic structure, and formant distribution, while preserving naturalness. This mechanism can, therefore, make it much harder to remove the perturbations, prevent TTS models from aligning semantics

†: Equal contribution.

2158-3927/26/$31.00 ©2026 IEEE
DOI 10.1109/DSN69566.2026.00037

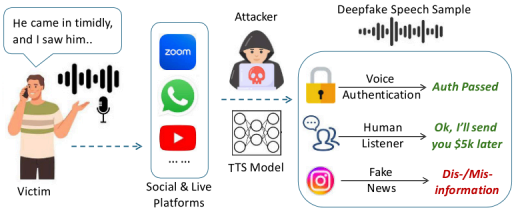


Fig. 1. Attackers steal victims' audio samples available online or infiltrate voice livestreaming platforms to train text-to-speech (TTS) synthesis models. The thus-synthesized voice samples are used for various malicious purposes, causing reputational and financial losses.

with the original timbre and hence impede intelligible voice synthesis. We further employ a speech encoder–decoder to shift optimization into the latent domain, and design a novel meta-perturbation mechanism to protect users in both real-time (e.g., livestreaming) and offline (e.g., social media) scenarios. Finally, inspired by adversarial training [10], [11], we introduce a loss function that aligns perturbations with TTS-salient features without requiring gradient access. Extensive experiments across models, datasets, and real-world setups demonstrate that SpeechShield achieves excellent robustness and effectiveness in ordinary and various adversarial scenarios, with perturbation generation up to 13× faster than SOTA methods, enabling its seamless integration in real time.

This paper makes the following main contributions:

- The first proposal of a universal and robust audio-adversarial perturbation defense, called SpeechShield, that preserves timbral naturalness and is readily transferable to both real-time and offline deployment scenarios.
- Development of a targeted embedding strategy that integrates perturbations into the user's core timbral features, making it difficult for TTS models to learn the protected audio while simultaneously resisting SOTA adversarial bypass techniques. Furthermore, lower generation latency and better preserved natural timbre are achieved by novel optimization in the compressed latent space with the "meta-perturbation" mechanism.
- Extensive experimentation across multiple TTS architectures, datasets, baseline defenses, and real-world microphone—speaker setups, empirically demonstrating SpeechShield's superior speech protection.

II. Background

We briefly introduce the stages of generating predicted audio within a speech synthesis model, along with the key acoustic features of speech.

A. Voice Synthesis

Speech synthesis technologies can be divided into two main categories: Text-to-Speech (TTS) (or Voice Cloning) and Voice Conversion (VC). Modern TTS systems first map input text to a target speaker's waveform by extracting and aligning acoustic features (e.g., phonemes, Mel-spectrograms, fundamental frequency F_0 and/or their embeddings) from recordings and reconstructing the speech signal. In contrast, voice conversion alters the timbre of an existing utterance while preserving its original linguistic content.

TTS has drawn widespread interest because it can synthesize speech from arbitrary text; TTS models are the focus of this paper. Modern TTS systems typically generate speech in two stages: (1) **Acoustic feature prediction.** A model converts input text embeddings to frame-level representations, such as Mel-spectrograms, F_0 contours. Some models (e.g., VALL-E 2 [12]) even compress such representations into discrete tokens. Early systems use autoregressive decoders with attention (e.g., Tacotron 2 [13]), while recent duration-based architectures (e.g., FastSpeech 2 [14] and StyleTTS [1]) deliver faster, more stable inference. (2) **Neural vocoding.** A vocoder transforms the predicted acoustic features to a time-domain waveform. SOTA neural vocoders (e.g., WaveNet [15], HiFi-GAN [16], WaveGlow [17], BERT-VITS2) learn a direct inverse mapping from spectrograms to audio, implicitly modeling the fine-grained periodicity, noise components, and phase information required for natural speech. During inference, textual inputs are first converted to linguistic features or phoneme-based representations. Then, the model can fit a speech sample that matches the speaker's voice characteristics based on the linguistic representation of the text.

B. Acoustic Features of Speech

Speech signals are characterized by three features governing timbre, prosody, and intelligibility: (1) **Fundamental frequency (F_0).** It reflects vocal-fold vibration rate and perceived as pitch, F_0 typically spans 80–300 Hz in adults (males: 85–180 Hz; females: 165–255 Hz) [18]. F_0 contours convey emotion and lexical meaning; accurate modeling is therefore critical for naturalness and speaker identity. (2) **Harmonic structure.** Periodic excitation generates overtones at integer multiples of F_0. These evenly-spaced bands enhance vowel clarity and timbre. Attenuation of upper harmonics degrades resonance and perceived naturalness. (3) **Formant patterns.** Formants are vocal tract resonances defining vowel identity. The first three peaks (F_1 (300–900 Hz), F_2 (800–2,500 Hz), and F_3 (2,000–3,500 Hz)) contribute to vowel quality the most and are essential for intelligibility and speaker recognition.

III. Threat Model

This section presents our defense scenario, system goal, and assumptions, as well as problem formulation.

A. Defense Scenario and System Goal

We consider a voice cloning/deepfake threat in which a user (**victim**) shares speech content on social media platforms or participates in real-time voice communication, such as livestreamed public speaking. Before the speech data reaches any external network or storage service, it goes through a protection mechanism deployed by the **defender**. This mechanism operates as a local background service. In real-time or live streaming scenarios, the defender may slice the incoming audio into segments using a sliding time window and add perturbations to each segment individually. In non-real-time or

offline scenarios, the defender processes the entire utterance as a whole. In both cases, the raw audio is transformed to adversarial examples before leaving the user's device. At the same time, an **adversary** may attempt to obtain the user's voice data through various means even if the data has been protected. For example, adversaries may use web crawlers to collect publicly available recordings or intercept live voice streams to extract audio samples. These collected samples can then be used to train or fine-tune open-sourced or commercial TTS models in order to clone the victim's voice. The cloned voice may be exploited in unauthorized activities such as spreading misinformation or conducting voice phishing.

The main system goal is to ensure that the adversary's TTS model trained on protected audio samples generates a speech which is semantically corrupted or timbrally distorted, rather than simply reducing its similarity to the victim's. This distortion must render the cloned speech unusable for impersonation. Furthermore, the added perturbations must be robust, meaning they should remain effective even when adversaries apply common purifications like speech denoising, spectral filtering, or adaptive retraining. The perturbations must also be perceptually friendly, meaning they should not interfere with human listeners' comprehension of the speech content or perception of the speaker's natural timbre. Finally, the perturbation generation must meet the latency constraints required for both real-time and offline use cases.

B. Assumptions on Adversaries and Defender

We assume the adversaries and the defender have the following capabilities for the design of a protection pipeline.

Adversaries. Adversaries are assumed to be able to acquire a user/victim's speech samples by crawling social media platforms or recording audio from the user's live speaking sessions. Using these samples, an adversary may train or fine-tune publicly available SOTA TTS models on commodity hardware for cloning the victim's voice (*Adversary 1*).

A more capable adversary may recognize that the audio contains protective perturbations and applies techniques such as denoising, replay-based purification, and data augmentation strategies to bypass or weaken the protection (*Adversary 2*).

An advanced adversary may have knowledge of the defender's surrogate model (e.g., code and parameters) but lacks specific optimization strategies used by the defender (e.g., loss functions). They may attempt to reconstruct clean speech by inferring these strategies or exploiting surrogate supervision, such as querying speaker recognition systems. Given the availability of open-source TTS models, this assumption is reasonable for such a sophisticated adversary. (*Adversary 3*).

Defender. The defender is assumed to have no access to the adversary's TTS model, training data/methodology. So, the defender can only apply perturbations directly to the original audio before leaving the user's device. However, the defender may leverage publicly available TTS models and record the victim's voice samples to form training datasets, and further guide the generation of perturbations during the design and training phases. However, the attacker does not know any details of the pre-trained surrogate TTS model, including but not limited to training datasets and parameters when training. Also, due to the nature of a black-box model, the attacker is not required to access the gradient within the selected surrogate TTS model.

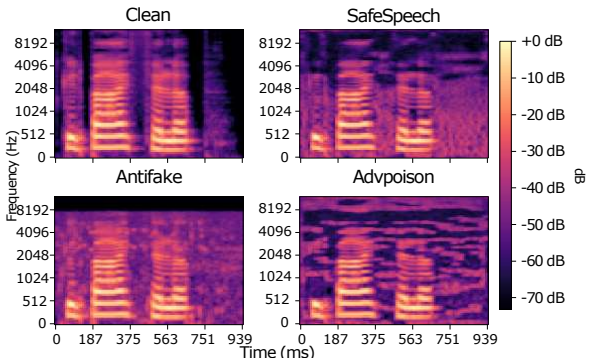


Fig. 2. Mel-spectrogram of adversarially protected audio generated by different methods for the "Good-by Philip!" speech sample.

C. Problem Formulation

We formulate the task of generating voice-protective perturbations as a constrained optimization problem. Let $x = [x_1, x_2, \ldots, x_n] \in [-1, 1]^n$ denote a clean audio waveform of length n. We seek a perturbed waveform $\tilde{x} = x + \delta \in [-1, 1]^n$ that simultaneously [2], [3]:

- makes an adversary's TTS model $G(\cdot)$ unable to generate high-quality cloned voice samples (*effectiveness*), measured by $\mathcal{L}_e\big(G(\tilde{x}), x\big)$;
- makes it hard for the adversary to circumvent protection via possible adversarial fine-tuning and denoising attempts (*robustness*), measured by $\mathcal{L}_r\big(G(\tilde{x}), x\big)$;
- preserves naturalness and intelligibility (*perceptual quality*), measured by $\mathcal{L}_g(\tilde{x})$.

IV. Design Details

This section details the design of our proposed pipeline.

A. Effective and Robust Voice Protection

Our objective is to generate adversarial perturbations that prevent SOTA TTS models from accurately learning and reproducing a user's voice. To this end, we first evaluate the protection efficacy of several existing methods [2], [3], [19]. Following their respective default experimental configurations, we generate adversarially protected speech samples using the LibriTTS dataset, focusing on speaker ID 5339. We adopt Bert-VITS2, a VITS-based end-to-end (e2e) TTS model as the surrogate model, to train and synthesize the voice of the target speaker. The synthesized speech is evaluated using word error rate (WER) and speaker similarity metrics.

Our experimental results indicate that the methods proposed in [2] and [19] fail to prevent the model from generating high-fidelity speech that closely resembles the target speaker's voice, with only minor background noise present. The speaker similarity scores for these two methods are 52.4% and 49.6%,

and their WERs are 26.1% and 27.3%, respectively. In contrast, the method in [3] produces speech heavily corrupted by Gaussian noise, making the content largely unintelligible. This is indicated by a WER of 91.2% and a significantly reduced speaker similarity score of 0.206. However, when we apply Demucs v3 [20] as a baseline speech denoising model to assess the robustness of these adversarial perturbations, we find that the perturbations generated by those three methods are removed. The WERs of denoised samples from [2], [19] and [3] are 23.7%, 26.2% and 31.9%, respectively. The Mel-spectrograms of the denoised speech closely resemble those of the original, unperturbed samples, and the TTS model is able to synthesize speaker-like speech with minimal residual noise. This demonstrates that SOTA voice protection strategies are ineffective in the standard adversarial robustness settings.

To further investigate the underlying cause behind the experimental result, we visualize the Mel-spectrogram of a protected speech sample containing the phrase "Good-bye, Philip!", as shown in Fig. 2. While the spectrogram reveals that existing methods do introduce certain irregular distortions (see diffused purplish shadows), the core acoustic characteristics of the original speech, including the fundamental frequency and harmonic structures (i.e., alternating light and dark bands) remain largely intact. Notably, although AdvPoison [19] induces distortions in higher-order harmonic components (e.g., those corresponding to frequencies around 3,600 Hz), these spectral regions lie well beyond the F_0 to F_3 formant bands and thus have negligible impact on the speaker's perceived timbre. Since timbral features are crucial for both speech denoising and TTS systems, we argue that the inherent vulnerability of SOTA voice protection stems from the weak coupling between the injected perturbations and these critical acoustic features.

Based on the above observation, we propose that coupling perturbations with timbral characteristics should be a key objective. By injecting structured noise to aperiodically distort harmonic energy and increase spectral flatness (specifically in F_1–F_3), one can effectively inject broadband "structured" noise into voiced frames. This hinders vowel distinction, causing the TTS model to mislearn the speaker's timbral alignment. To achieve this goal, we define two key loss functions: the harmonic energy loss $\mathcal{L}_h$ and the spectral flatness loss $\mathcal{L}_f$. Both losses are intended to disrupt the harmonic structure and its energy distribution, which are crucial for TTS models to reproduce speaker-specific characteristics. However, one of the main challenges in optimizing these objectives is accurate estimation of spectral characteristics of fundamental frequency and harmonics in a differentiable manner. Conventional methods for pitch tracking (e.g., YIN [21] and SWIPE) employ discrete optimization techniques, including local min-value search and thresholding, which result in discontinuous outputs. These methods are inherently non-differentiable and hence incompatible with the gradient-based adversarial optimization. To overcome this limitation, we design a fully differentiable pipeline for estimating the fundamental frequency contour of a speech signal. Based on this contour, we identify the spectral bin indices corresponding to integer multiples of the estimated F_0 (i.e., harmonic components) and accumulate the spectral energy at these locations. The total harmonic energy is then averaged across all frames, forming a loss term that enables the perturbation to distort the harmonic structure of the speech sample.

Given a speech sample segmented into L frame windows, let $x_l[n]$ be the n-th sample of the l-th frame, where $n = 0, 1, \ldots, N-1$. We first compute the autocorrelation function across all lag values τ for each window:

$$r_{tr}(l,\tau) = \sum_{n=0}^{N-\tau-1} x_l[n]\, x_l[n+\tau] \quad (\tau = 0, 1, \ldots, \tau_{\max}), \quad (1)$$

where the valid pitch range is bounded by minimum and maximum lags: $\tau_{\min} = \lfloor \frac{f_s}{f_{0,\max}} \rfloor$,$\tau_{\max} = \lfloor \frac{f_s}{f_{0,\min}} \rfloor$, and f_s is the sampling rate, $f_{0,\min}$ and $f_{0,\max}$ are the minimum (typically 50Hz) and maximum (typically 500Hz) [22] expected F_0 values, respectively. We discard autocorrelation values outside this lag interval to form a truncated autocorrelation sequence. To obtain a differentiable estimate of the frame-level pitch, we apply a soft-argmax over the truncated autocorrelation values:

$$w(l,\tau) = \frac{\exp\left(r_{\text{tr}}(l,\tau)\right)}{\sum_{\tau'=\tau_{\min}}^{\tau_{\max}} \exp\left(r_{\text{tr}}(l,\tau')\right)}, \hat{\tau}(l) = \sum_{\tau=\tau_{\min}}^{\tau_{\max}} \tau \cdot w(l,\tau). \quad (2)$$

The estimated fundamental frequency for frame window l is then computed as $\hat{f}_0(l) = \frac{f_s}{\hat{\tau}(l)}$. Using this continuous pitch contour, we identify harmonic frequencies by computing the spectral bin indices corresponding to integer multiples of $\hat{f}_0(l)$: $b_k(l) = \lfloor k\hat{f}_0(l)/(f_s/N) \rfloor, \quad k = 1, 2, \ldots, K$. This facilitates the computation of the harmonic energy loss, promoting the destruction of energy in the harmonic bands. Since the harmonic envelope spectrum usually exhibits low-pass characteristics, and the energy decays (approximately) exponentially with the harmonic order [22]. In order to destroy its distribution, we define the final formulation of $\mathcal{L}_h$ as:

$$\mathcal{L}_h = -\log\left(\frac{1}{L}\sum_{l=1}^{L}\sum_{k=1}^{K} M(b_k(l), l)\right) \quad , \quad (3)$$

where $M(f,l)$ denotes the spectral magnitude at frequency bin f in frame window l, calculating by short-time Fourier transform (STFT).

While the harmonic energy loss $\mathcal{L}_h$ effectively targets energy concentrations at integer multiples of the fundamental frequency, the corresponding perturbations are inherently localized at narrow spectral bands. Our empirical analysis shows that although this local interference can degrade some timbre features, SOTA denoising models (such as Demucs v3) can use the correlations learned in the spectral domain to reconstruct fundamental frequencies and harmonics. So, in practice, the protection effect of $\mathcal{L}_h$ alone is limited. To address this limitation, we propose the spectral flatness loss $\mathcal{L}_f$, which promotes the injection of broadband perturbations distributed across harmonic regions. Rather than merely suppressing spectral peaks, this objective helps us reduce the contrast between harmonic and non-harmonic components, thus diminishing the

perceptual and structural cues essential for speaker identity reconstruction. To formalize this, we propose the Spectral Flatness Measure (SFM), defined as the ratio of the geometric mean (GM) to the arithmetic mean (AM) of the spectral magnitudes within each frame:

$$GM(l) = \exp[\frac{1}{F}\sum_{f=1}^{F}\log M(f,l)], AM(l) = \frac{1}{F}\sum_{f=1}^{F} M(f,l), \tag{4}$$

$$SFM(l) = \frac{GM(l)}{AM(l)}, \tag{5}$$

where in Eq. (3) we achieve the numerically stable computation by leveraging the following equivalence between logarithmetic and exponential operations: $\exp(\frac{1}{F}\sum \log x_i) = (\prod_i x_i)^{1/F}$. Intuitively, the arithmetic mean (AM) reflects overall energy and is highly sensitive to strong spectral peaks, such as harmonics. In contrast, the geometric mean (GM) penalizes low-magnitude values more heavily, being suppressed by valleys in the spectrum. When the spectral distribution exhibits high peak-valley contrast, AM becomes large while GM remains small, resulting in low spectral flatness. As energy becomes more evenly spread, SFM increases and approaches 1 when all bins are equal in magnitude. This aligns with the AM-–GM inequality, guaranteeing $SFM \leq 1$, with equality holding only under rigorous spectral uniformity. Thus, we define the spectral flatness loss as:

$$\mathcal{L}_f = \frac{1}{L}\sum_{l=1}^{L}(1 - SFM(l)). \tag{6}$$

So far, we have designed the main mechanisms for introducing "structured" perturbations. To further optimize the coupling between the perturbation and features learned by TTS and ensure that these perturbations *universally* degrade performance across different TTS models, it is important to guide the optimization based only on the input and output features of the model, without relying on access to model-specific gradients. We leverage the observation that well-trained genereative models produce outputs matching training data distributions [23]. Moreover, inspired by adversarial training [10], [19], [1], we hypothesize that for a given speech sample x and a corresponding perturbation δ, if the perturbation contains those features that the TTS model considers relevant to the phoneme, applying them should help the model better fit the protected sample's distribution; whereas irrelevant features (e.g., noise) hinder alignment. Thus, letting $\hat{x}_1$ and $\hat{x}_2$ denote generator outputs for clean (x) and protected ($x+\delta$) samples, and $D(\cdot,\cdot)$ a discrepancy function. A well-optimized perturbation should satisfy:

$$D(x+\delta, \hat{x}_2) \leq D(x, \hat{x}_1). \tag{7}$$

In voice-related tasks, the similarity between sample distributions can be measured by calculating the mel-spectrogram discrepancy. At the same time, we finally apply $\mathcal{L}_h$ and $\mathcal{L}_f$ to $\hat{x}_2$ to make the perturbation weaken the intelligibility of the generated audio by TTS, in terms of phoneme-related features.

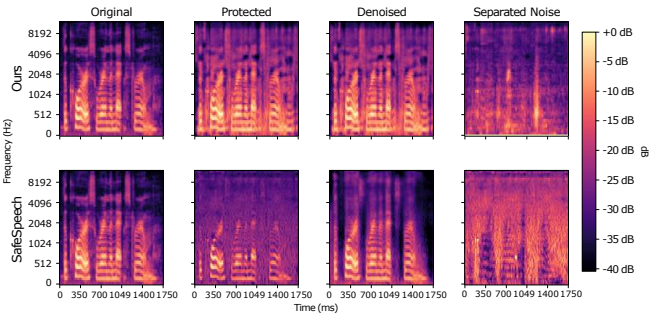


Fig. 3. Performance comparison of our proposed optimization objectives and Safespeech (SOTA method) in adversarial denoising for protected samples containing "The chorus was in the next room."

Therefore, the combined final optimization objective can be summarized as:

$$\mathcal{L}_r = \mathcal{L}_f + \mathcal{L}_h, \tag{8}$$

$$\mathcal{L}_e = D(x+\delta, \hat{x}_2) - D(x, \hat{x}_1). \tag{9}$$

To evaluate the effectiveness of the proposed optimization objectives for adversarial audio generation, we conduct a case study using a speech sample from speaker ID 3922 in the LibriTTS dataset, containing the phrase "The chorus was in the next room." Our perturbation objectives (Eq. (9)) are applied to generate a protected version of the speech and its robustness is compared against speech denoising with that of SafeSpeech, a prominant SOTA method. The parameters used for optimization are the same as in Section V-A. As shown in Fig. 3, after processing with Demucs v3, our protected sample retains a substantial portion of the perturbation energy (primarily concentrated in the harmonic wide-band regions) without degrading or compensating for the speech-relevant components. In contrast, SafeSpeech attempts to impose Gaussian-distributed noise to hide the original speech, which is largely eliminated by the denoiser, allowing the original speech characteristics to be almost fully restored.

B. Optimization of Disturbance Generates Delay and Voice Naturalness

While the constraints introduced in Section IV-A steer the optimization toward the generation of impactful adversarial perturbations, similarly to [3], [19], [24], SpeechShield operates directly in the audio waveform domain. Given the high sampling rates typically used in speech processing (e.g., 24kHz or 48kHz), even short audio segments consist of large data vectors with significant redundancy points. Performing optimization in such a high-dimensional space is computationally expensive. For instance, [24] reports that generating a 15-second protected sample takes approximately 300 seconds, while [3] requires about 23 seconds/180 steps to protect a 5-second sample using an NVIDIA A800 GPU. Such high latencies make them impractical for time-sensitive applications like live streaming.

Moreover, when perturbations are added directly in the time domain, the optimization of the phase spectrum is often ignored, introducing unnatural audio artifacts. While multiplica-

Algorithm 1 Generate Meta-Perturbation for the Speaker

Require: Proxy data samples $\mathcal{Y}$, initialized meta-perturbation θ, step size α, meta-learning rate β, inner-loop steps K, training iterations M, batch size N, optimization target $\mathcal{L}$.

for $i = 1, \ldots, M$ **do**
  while not all batches have been sampled **do**
    Sample a batch $\{y_1, \ldots, y_N\}$ from $\mathcal{Y}$
    for $j = 1, \ldots, N$ **do**
      Initialize perturbation $\delta_j^{(0)} = \theta$
      for $k = 1, \ldots, K$ **do**
        $\mathcal{L}_{T_j} = \mathcal{L}(y_j + \delta_j^{(k-1)})$
        $\delta_j^{(k)} = \mathrm{Proj}_{\|\cdot\|_\infty \le \epsilon}\left(\delta_j^{(k-1)} - \alpha \nabla_\delta \mathcal{L}_{T_j}\right)$
      end for
      $y_j = y_j + \delta_j^{(K)}$
    end for
    Meta-update: $\theta \leftarrow \theta + \beta\left(\frac{1}{N}\sum_{j=1}^{N}(\delta_j^{(K)} - \theta)\right)$
  end while
end for
return θ

tive perturbations [5] improve alignment, they remain bound by the waveform domain's high dimensionality. To address this, we propose mapping the search space via $f : \mathcal{X} \to \mathcal{Y}$ to a compact feature space $\mathcal{Y}$ that preserves essential structure and implicit phase. We implement this using the SeaNET encoder-decoder [25], which enables differentiable reconstruction and inherently captures phase, obviating the need for manual estimation. This intermediate space should preserve the essential components required for faithful waveform reconstruction, while reducing redundancy and incorporating implicit phase information. We also employ a differentiable STOI loss $\mathcal{L}_g$ [26] to prevent over-optimization and maintain the timbral fidelity of the audio.

Despite the benefits of a compact representation space, traditional optimization methods (e.g., PGD) still suggest a random initialization of latent perturbations for each input sample, often following a Gaussian distribution [27]. This cold-start approach remains time- and resource-ineffective. To address this problem, we adopt a meta-learning strategy that enables rapid adaptation across speech samples. Specifically, we learn a meta-perturbation θ in the feature latent space that targets generalizable vocal patterns related to the victim, such as the general distribution of harmonic energy. When presented with a new speech sample, this meta-perturbation serves as a strong initialization that can be fine-tuned with a few steps, to generate the corresponding protected audio representation. Formally, for a speech latent sample $y_i \in \mathcal{Y}$, where $\mathcal{Y}$ represents the latent representation of the victim's surrogate speech dataset. We initialize its perturbation with a randomly initialized meta-perturbation parameter θ: $\delta_i^{(0)} \leftarrow \theta$. We then apply K steps of Projected Gradient Descent (PGD) to minimize the composite speech protection loss $\mathcal{L} = \mathcal{L}_r + \mathcal{L}_e + \mathcal{L}_g$

Algorithm 2 SpeechShield Pipeline

Require: $(x, \text{text}) \in \mathcal{X}$, encoder E, decoder D, surrogate TTS model G, meta-perturbation θ, fine-tune steps N.

Params: perturb boundary ϵ, weight coefficients α, β, γ, learning rate λ.

Output: perturbed audio sample $\tilde{x}$.

$y \leftarrow E(x)$, $\delta \leftarrow \theta$,
for $i = 1$ to N **do**
  $y' \leftarrow y + \delta$;
  $\tilde{x} \leftarrow D(y')$;
  $\hat{x}_1, \hat{x}_2 \leftarrow G(\tilde{x}, x, \text{text})$;
  $C_1 \leftarrow \mathcal{L}_r(\hat{x}_2, x)$, $C_2 \leftarrow \mathcal{L}_e(\hat{x}_1, \hat{x}_2, x)$, $C_3 \leftarrow \mathcal{L}_g(\tilde{x})$;
  $\delta \leftarrow \mathrm{Clamp}(\delta - \lambda \nabla_\delta(\alpha C_1 + \beta C_2 + \gamma C_3), -\epsilon, \epsilon)$;
end for
return $\tilde{x}$.

for its decoded voice, with learning rate α:

$$\delta_i^{(k+1)} = \mathrm{Proj}_{\|\cdot\|_\infty \le \epsilon}\Big(\delta_i^{(k)} - \alpha \nabla_\delta \mathcal{L}\big(G(D(y_i + \delta_i^{(k)})), D(y_i + \delta_i^{(k)})\big)\Big), \tag{10}$$

where $G(\cdot)$ refers to the surrogate TTS model, and $D(\cdot)$ is the decoder of SeaNET. After K steps, we obtain a sample-specific perturbation $\delta_i^{(K)}$. To improve generalization, we update the meta-perturbation θ toward the average of task-specific perturbations across a batch of N samples (the average of perturbation is approximately equivalent to the average of gradients in direction under the first-order approximation). The meta-update rule is defined as $\theta \leftarrow \theta + \beta\left(\frac{1}{N}\sum_{i=1}^{N}(\delta_i^{(K)} - \theta)\right)$, where β is the meta-learning rate. This meta-update step improves from the first-order Reptile algorithm [28] to evolve toward a universal initializer that adapts efficiently across new speech inputs. The full optimization procedure is presented in the extended version of this paper [1]. The meta-perturbation generation algorithm is shown in Algorithm 1, and the overall perturbation pipeline is shown in Algorithm 2.

V. Evaluation

A. Experimental Setup

Prototype. We have implemented SpeechShield on the PyTorch platform and generated the perturbation using one NVIDIA A800 GPU. We set the default parameter configuration as $\alpha = 1.0, \beta = 0.4, \gamma = 0.1, k = 6, \lambda = 1e^{-3}, \mu = 2e^{-4}, \epsilon = 0.3$, meta inner-loop steps $K = 180$, and fine-tune steps $N = 10$. We set meta-perturbation training batch size $T = 27$. When training the encoder-decoder of SeaNET, we use LibriTTS as the training dataset and adopt MSE and STFT losses. The default adversary is *Adversary 3*. We adopt BERT-VITS2 as the surrogate TTS model, as it is representative of today's TTS common design space (text→prosody→speaker→neural decoder). Besides, our

[1] https://tinyurl.com/5fr9kfte.

TABLE I
COMPARISON OF EFFECTIVENESS AMONG DIFFERENT TTS MODELS AND PROTECTION METHODS

Method	BERT-VITS2			VITS			GlowTTS			StyleTTS2			MB-iSTFT-VITS			Avg		
	MCD	WER	SIM	MCD	WER	SIM	MCD	WER	SIM	MCD	WER	SIM	MCD	WER	SIM	MCD	WER	SIM
GT†	–	16.37	–	–	16.37	–	–	16.37	–	–	16.37	–	–	16.37	–	–	16.37	–
RN†	5.45	27.75	0.47	5.50	33.27	0.45	9.85	49.68	0.31	5.46	23.62	0.47	5.40	24.65	0.49	6.33	31.79	0.44
AP† **[19]**	10.47	57.70	0.52	8.59	50.70	0.40	13.00	94.77	0.19	9.31	27.77	0.35	8.07	35.22	0.39	9.89	53.23	0.37
SEP†**[29]**	8.37	57.92	0.32	8.00	55.92	0.29	14.25	81.75	0.21	7.21	26.00	0.32	7.64	62.63	0.27	9.09	56.84	0.28
PTA† **[30]**	11.19	59.04	0.29	10.04	72.30	0.24	17.89	85.02	0.14	9.69	26.85	0.25	9.76	54.70	0.24	11.71	59.58	0.23
AF† **[2]**	7.74	48.97	0.25	6.16	63.60	0.22	12.34	98.41	0.09	7.76	42.89	0.21	6.75	58.42	0.23	8.15	62.46	0.20
SS(w/o.P)†	15.01	99.15	0.18	11.60	93.29	0.18	22.09	102.41	0.08	9.16	55.16	0.13	13.77	94.75	0.16	14.33	88.95	0.15
SS(w.P)†	15.55	97.11	0.29	12.95	91.82	0.20	22.45	96.93	0.16	10.83	25.84	0.37	10.05	88.15	0.19	14.37	79.97	0.24
Ours†	**11.79**	**94.78**	**0.11**	**12.90**	**138.41**	**0.02**	**18.83**	**103.27**	**0.03**	**8.98**	**56.62**	**0.15**	**9.94**	**91.13**	**0.12**	12.49	**96.84**	**0.09**
GT‡	–	8.06	–	–	8.06	–	–	8.06	–	–	8.06	–	–	8.06	–	–	8.06	–
RN	6.01	26.33	0.57	6.17	16.49	0.52	9.59	37.92	0.34	6.28	10.50	0.47	6.10	11.79	0.47	6.83	20.61	0.47
AP‡ **[19]**	10.44	37.92	0.40	9.29	52.71	0.35	14.21	87.32	0.07	10.26	13.78	0.29	9.15	28.34	0.36	10.67	44.01	0.29
SEP‡ **[29]**	8.28	50.57	0.43	8.77	46.41	0.34	14.42	76.62	0.12	8.41	14.35	0.29	8.39	32.62	0.32	9.65	44.11	0.30
PTA‡ **[30]**	11.50	46.62	0.37	12.04	57.05	0.28	17.84	82.88	0.08	9.47	15.96	0.37	11.04	29.44	0.25	12.38	46.39	0.27
AF‡ **[2]**	10.12	66.68	0.30	9.27	59.84	0.24	10.99	91.36	0.12	5.19	35.82	0.27	7.22	34.36	0.29	8.56	57.61	0.24
SS(w/o.P)‡	15.90	89.49	0.32	14.35	54.14	0.43	20.32	99.71	0.04	10.67	31.30	0.04	14.52	68.91	0.35	15.15	68.71	0.24
SS(w.P)‡	15.27	81.57	0.31	14.53	56.19	0.48	19.36	99.37	0.02	11.64	27.84	0.05	14.21	62.44	0.34	15.00	65.48	0.24
Ours‡	**13.26**	**120.72**	**0.21**	**12.71**	**75.36**	**0.28**	**17.41**	**99.66**	**0.02**	**11.11**	**46.07**	**0.01**	**12.67**	**63.66**	**0.22**	13.43	**81.09**	**0.15**

†: results on D_1 (LibriTTS); ‡: results on D_2 (CMU ARCTIC); **GT**: Ground-truth; **RN**: Random Noise; **SS(w/o.P)**: SafeSpeech **without** features on human perceptual optimization; **SS(w.P)**: SafeSpeech **with** features on human perceptual optimization; Higher MCD is better, higher WER (%) is better, and lower SIM is better.

optimization focuses on model-agnostic acoustic attributes, making it applicable to transfer across different TTS backbones (See Table I for evaluation details). For real-world scenarios, we conduct tests using the built-in microphone of Apple MacBook Air M3, Newmine ZM01, AirPods 4 (with active denoising feature), and Philips SHM1900 as external microphones. We play the perturbed voices through built-in speaker of Apple MacBook Air M3, Dell Inspiron 3501, as well as JBL Flip 6 and Xiaomi Portable Mini as external speakers.

Datasets. We assess SpeechShield on two standard speech datasets used by speech-related researchers: LibriTTS [31] and CMU Arctic [32]. The LibriTTS dataset contains 585 hours of sentence-segmented read speeches by multiple distinct English speakers at a 24kHz sampling rate, while CMU Arctic comprises 4 single-speaker databases of phonetically balanced English, and is recorded at a 16 kHz sampling rate. Unless noted otherwise, we split each $\geq$150-utterance speaker 80/20(train/text), fine-tune TTS models on protected train audio, synthesize the held-out test texts, and compute WER/SIM/MCD/VN on the synthesized audio against the clean references, with denoising and mixed-data ablations following the same protocol.

Evaluation Metrics. We adopt the following metrics when conducting the evaluation: **(1) Word Error Rate (WER)** quantifies transcription discrepancy using Whisper [33]. It is calculated as $WER = \frac{S+D+I}{N}$, where S, D and I denote the number of substitutions, deletions and insertion errors of words, and N is the total number of words. Higher WER values reflect noisier, less intelligible speech; **(2) Signal-to-Noise Ratio (SNR)** measures the energy ratio of speech x to perturbation δ : $SNR = 10\log\left(\|x\|_2^2/\|\delta\|_2^2\right)$ (dB). Lower values reflect more intrusive perturbations. **(3) Speaker Similarity (SIM)** measures identity preservation via cosine similarity between embeddings extracted using ECAPA-TDNN [34]. Lower scores indicate stronger protection. **(4) Mel-Cepstral Distortion (MCD)** quantifies spectral envelope difference. We can get MCD by calculating DTW-aligned mel-spectral coefficients [35]. **(5) Real-time Coefficient (RTC)** quantifies generation efficiency as $RTC = T_f/T_{audio}$, where T_f is the average time used for generating the perturbed audio and T_{audio} is audio duration. $RTC < 1$ ensures no user-perceptible pause. **(6) Voice Naturalness (VN)** is evaluated objectively using UTMOSv2 [3], where higher values indicate better voice quality.

Baselines. We compare SpeechShield with 6 existing voice protection techniques: AdvPoison [10], SEP [29], PTA [30], AttackVC [36], AntiFake [2], and SafeSpeech [3]. All of these baselines are evaluated based on their best configurations given in their instructions. Also, we select SOTA TTS models BERT-VITS2 [37], MB-iSTFT-VITS [38], VITS [39], GlowTTS [40], and StyleTTS2 [41], and use the protected audio to fine-tune these models, evaluating the applicability of the perturbations to different models.

B. Effectiveness Comparison

We compare SpeechShield with 6 existing speech protection methods, and present the evaluation results in Table I. For SafeSpeech [3], we list the results both with the perceptual-optimization loss enabled (default setting) and with that loss disabled. The results show that SpeechShield achieves effective protection on both datasets and generalizes well across different TTS models. At the same time, SpeechShield generally outperforms SOTA (SafeSpeech with features on human perceptual optimization enabled) under multiple TTS models. SpeechShield achieves maximum WERs of 138.41 and 120.72 on LibriTTS and CMU Arctic, respectively. Meanwhile, speaker similarity (SIM) drops to as low as 0.01, indicating that SpeechShield-protected data can dramatically lower the intelligibility and perceptual quality of synthe-

TABLE II
EVALUATION METRICS FOR VARYING ϵ.

ϵ	SIM	WER (%)	MCD	VN(↑)
0.1	0.24	76.87	9.82	2.52
0.2	0.15	84.83	10.34	2.29
0.3	0.11	94.78	11.79	2.12
0.4	0.09	97.12	12.96	1.99
0.5	0.06	100.06	16.84	1.82
0.6	0.02	102.97	18.09	1.69
Baseline	0.47	27.75	5.45	3.23

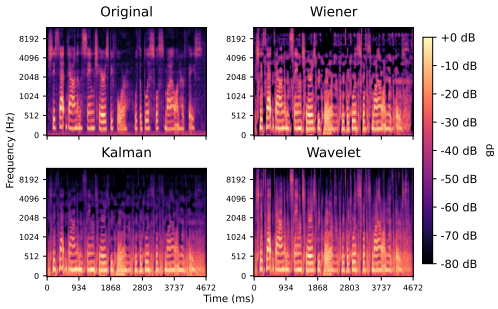


Fig. 4. Denoising spectrum results for a sample containing "She sat down in the same chair as before, with a blissful smile on her face", under different signal processing methods.

sized speech. We also evaluate the impact of varying the perturbation-amplitude constraint ϵ on the naturalness of the protected audio and the quality of the synthesized output (Table II). As ϵ increases, WER and MCD both rise while SIM steadily decreases, implying that looser constraints yield stronger protection but at the cost of lower voice naturalness (VN). In practice, users must carefully tune ϵ to balance between perceptual quality and protection strength. Even under the tightest constraints, SpeechShield outperforms the baseline that uses random Gaussian noise by a wide margin.

C. Robustness Against Adversarial Denoising

Any protective perturbation that remains within the audible spectrum can be detected and exploited by an adversary. To recover and generate clean speech, an attacker may deploy various adversarial denoising techniques. In what follows, we evaluate SpeechShield's robustness against several common adversarial denoising techniques. All the following experiments are conducted on the LibriTTS dataset with the BERT-VITS2 model.

Denoising Based on Signal Processing. We evaluate robustness against Wiener filtering, Kalman filtering, and wavelet thresholding. Table III shows SpeechShield outperforms existing approaches across all three, specifically achieving a peak WER of 116.23% and SIM of 0.14 under wavelet thresholding. While matching SOTA on Kalman filtering, our method increases WER by ≈28.5% over SOTA for Wiener and wavelet denoisers. Fig. 4 visualizes the results. This resilience stems from spectral assumption mismatches: Wiener filtering cannot suppress noise peaks coinciding with speech bands without attenuating the signal, while wavelet thresholding targets high frequencies but ignores the low-frequency content we exploit.

TABLE III
ROBUSTNESS AGAINST CLASSICAL DENOISING METHODS

	Kalman		Wavelet		Wiener	
Method	WER%	SIM	WER%	SIM	WER%	SIM
Ours	**93.12**	0.15	**116.23**	**0.14**	**80.15**	**0.22**
SafeSpeech	91.25	**0.13**	77.49	0.19	62.24	0.25
AntiFake	39.48	0.26	36.88	0.36	47.80	0.30
PTA	47.37	0.31	44.23	0.34	43.72	0.26

Denoising Based on Deep Learning (DL). DL-based denoising models forgo an explicit spectral analysis and instead learn a direct mapping from perturbed to clean speech distributions. In our experiments, we apply four SOTA approaches: Demucs v3 [42], AudioPure [43], MDX23C [44], BS-RoFormer [45] and PhonePuRe[46] to denoise the protected samples, and then train the TTS model using those samples. As shown in Table IV, SpeechShield outperforms the other methods against DL-based denoisers. When processed by Demucs v3, it achieves a WER of 92.88% and a SIM of 0.13, and after analyzing the separated noise, we find that Demucs v3 removes almost none of the injected perturbations. Under the diffusion-based AudioPure and PhonePuRe models, SpeechShield maintains WERs of 87.61%, 71.16% and SIMs of 0.18, 0.27, respectively; for MDX23C and BS-RoFormer, SpeechShield achieves WERs of 91.92% and 92.94%, while SIMs are 0.14 and 0.11, respectively, demonstrating that SpeechShield provides much better protection than SOTA solutions.

D. Resilience against Other Adversarial Tasks

Fine-tuning with Mixed Data. In large-scale cloning, an adversary may fine-tune a single TTS model on multiple speakers (1 protected, $n-1$ clean) to improve efficiency. We simulate this by jointly training on five speakers (one victim). As shown in Fig. 5, fine-tuning solely on the protected victim yields MCD=13.12, WER=95.21%, and SIM=0.13. Mixing in four clean speakers causes only minor performance shifts: MCD decreases by 0.91, WER drops by 13.41%, and SIM increases by 0.11. Protection remains strong, exceeding clean-data baselines. We conjecture that including other speakers allows the TTS model to learn generalized timbral patterns—partially realigning the victim's features—but fails to fully negate the adversarial perturbations.An adversary might also mix clean same-gender audio with the victim's denoised samples to recover timbre. We test this by treating the victim's denoised audio (AudioPure+MDX23C) and clean benign samples (added at 10%, 20%, 30%, 40%, and 50% ratios) as a single speaker during fine-tuning. Fig. 6 shows that while MCD and WER decrease as clean data increases, SIM1 (victim) and SIM2 (benign) converge at ≈0.15, and WER remains above 70%. This confirms that even with clean data augmentation, perturbations sufficiently interfere with the synthesis to prevent high-quality matching of the victim's timbre.

An attacker might adopt data augmentation to weaken protection by reducing the proportion or structure of per-

TABLE IV
ROBUSTNESS AGAINST DL-BASED DENOISERS

	Demucsv3		AudioPure		MDX23C		BSRoformer		PhonePuRe	
	WER	SIM	WER	SIM	WER	SIM	WER	SIM	WER	SIM
Ours	**92.88**	**0.13**	**87.61**	**0.18**	**91.92**	**0.14**	**92.94**	**0.11**	**71.16**	**0.27**
SS	51.16	0.30	66.12	0.28	49.83	0.34	57.71	0.32	45.48	0.39
AF	34.08	0.39	35.12	0.41	30.02	0.44	29.86	0.48	24.75	0.51
PTA	36.12	0.41	41.15	0.44	38.96	0.29	28.87	0.46	25.12	0.50

SS: SafeSpeech; **AF**: AntiFake.

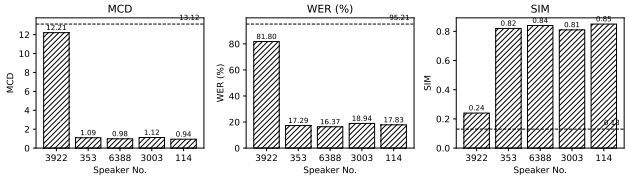


Fig. 5. Protection performance when fine-tuning on multiple speaker samples simultaneously.

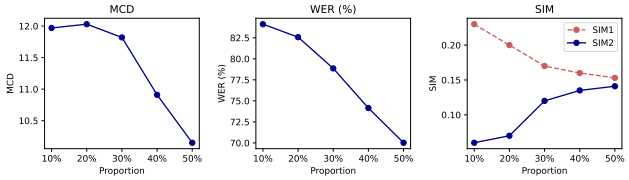


Fig. 6. Protection performance when the victim's protected audio samples are mixed with clean samples from other speakers, to create a single-speaker fine-tuning dataset.

turbed samples. We evaluate three strategies: (1) Upsampling/Downsampling — upsampling interpolates new points to dilute perturbations using methods like linear interpolation or windowed filters, while downsampling applies anti-aliasing filtering and discards samples unpredictably, disrupting patterns; (2) Lossy Compression — codecs remove or quantize perceptually redundant bands, potentially eliminating adversarial noise; (3) Speed Adjustment — stretching or compressing audio duration warps perturbation envelopes and periodic structures. We randomly select a speaker's audio samples and apply various augmentations during training. For resampling, we downsample to 14 kHz and upsample to 34 kHz during TTS training, while keeping inference at 24 kHz. For lossy compression, the audio samples are transcoded to the MP3 format before training. For speed adjustment, we use torchaudio to modify playback to 0.8× and 1.2× the original duration, respectively.

The evaluation results are listed in Table V; after applying augmentation methods, WER shows a slight decrease and SIM a slight increase compared to training on unaugmented samples, and the protection remains effective. This is because SpeechShield embeds the perturbation effectively within the speech's harmonic structure, and the above augmentations cannot effectively decouple the perturbation from the original audio.

Gradient-based Inversion. An intelligent adversary may guess possible optimization strategies and formulate an inverse

TABLE V
IMPACT OF DATA AUGMENTATION ON PROTECTION PERFORMANCE

Metric	US	DS	SA1	SA2	AC	Baseline
WER (%)	87.34	91.49	94.13	92.26	90.38	96.83
MCD	12.27	12.87	12.49	11.98	12.16	12.18
SIM	0.13	0.22	0.18	0.20	0.23	0.18

US: Up-sampling; **DS**: Down-sampling; **SA1**: 1.2x Speech Rate; **SA2**: 0.8x Speech Rate; **AC**: Audio Compression.

optimization by solving $\min_x \left\|x + \delta(x) - \tilde{x}\right\|^2$, to estimate the inversed gradient direction, "peel off" the perturbation and recover the original clean signal x. Moreover, the adversary may even query publicly available platforms that may have speaker recognition (SR) systems registered with the target speaker's identity, and exploit the recognition scores to guide the optimization direction, thereby improving the accuracy of recovery.

We consider using ECAPA-TDNN as the SR system, and train it with all the samples from speaker 3922 in LibriTTS (due to the largest number of samples), along with all samples from 50 randomly selected speakers, using the default configuration. Following [3], [9], we adopt Natural Evolution Strategies(NES) to compute the natural gradient estimate of the optimization objective $\mathcal{L}'(x)$ with respect to the distribution parameters: $\bigtriangledown_\theta J(\theta) = \mathbb{E}_{x\sim p_\theta}[\mathcal{L}'(x)\bigtriangledown_\theta \log p_\theta(x)]$, where we set $\mathcal{L}'(x)$ as the combination of the cosine distance of embedding vector with the target speaker, the Mel-spectrum similarity and the STOI score. As for NES, we set 50 samples per iteration and run 1,000 iterations, resulting in a total of 50,000 queries. After 100 optimization steps, we use the restored samples to fine-tune BERT-VITS2 and GlowTTS for speech synthesis. Compared to the perturbed samples (SIM=0.121/0.095, WER%=95.17/98.36), the samples generated through the gradient-based inversion method are observed to achieve SIM=0.173/0.144 and WER%=86.38/88.15, respectively, which are still far higher than the baseline of WER%=16.37. Although the adversary may have some estimation of the possible optimization direction, the estimated gradient carries less information than the true gradient, making it difficult to obtain the precise inverse direction of the original audio. Moreover, SpeechShield does not consider SR accuracy as an optimization objective, and hence introducing the surrogate SR model offers little help for the true gradient estimation.

Robustness in Real-world Scenarios. In real-world usage, users may record their voice via a microphone, apply the

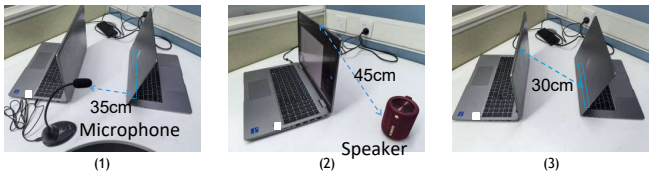


Fig. 7. Real-world experimental setup.

perturbation, and then upload the protected audio to the Internet. To evaluate the robustness of SpeechShield in real-world scenarios, we used diverse microphones (MacBook Air M3 built-in, Newmine ZM01, AirPods 4 with active denoising, Philips SHM1900) and speakers (MacBook Air M3, Dell Inspiron 3501, JBL Flip 6, Xiaomi Portable Mini). As shown in Fig. 7, we tested three configurations: (1) Laptop speaker to external microphone; (2) External speaker to laptop microphone; and (3) Laptop speaker to another laptop's microphone. For each configuration, we play a randomly selected speaker's all utterances from the LibriTTS dataset, capture the over-the-air audio via a microphone, and process it through our pipeline to generate the perturbations. We leverage the settings to mimic live deployment scenarios and time-sensitive applications like virtual meetings.

After aligning these recordings, we use them to train the TTS model. We report that the average WERs are 97.25%, 102.18%, and 95.83%; SIMs are 0.06, 0.08, and 0.02; and SNRs are -2.46dB, -2.52dB, and -2.60dB for the three configurations, respectively. The results show that our perturbation remains effective in realistic indoor environments, since it is tightly coupled with salient speech features (e.g., harmonics) that typical room acoustics do not significantly erase or distort.

Extended Exploration: Robustness in Voice Conversion. Voice conversion (VC) poses a similar threat to TTS, using a single reference sample to transfer a victim's timbre to get a target audio. To assess SpeechShield against VC, we use Seed-VC with pretrained weights on LibriTTS pairs (one perturbed reference, one source). Averaged over 10 trials, we found that standard perturbations yield weak protection, with WER=31.63% and SIM=0.79, as VC models rely on time-pooled speaker embeddings rather than the frame-level content targeted by SpeechShield. However, we found that adding a contrastive speaker embedding loss $L_{sv}(x) = \max(0, \cos(e(\tilde{x}), e(x)) - m_{neg})$ (with a coefficient of 0.1 and $m_{neg} \in [0, 0.2]$) via an ECAPA encoder [36] significantly improves protection. This yields WER=78.12% and SIM=0.19, preventing the capture of the victim's timbre. While effective, optimizing for speaker embedding distance is technically distinct and remains beyond this paper's scope.

E. Latency Performance Comparison

SpeechShield replaces the random initialization used in prior methods with a learned "meta-perturbation" as the starting point for optimization and employs an encoder–decoder architecture to fine-tune individualized perturbations in a compressed latent space. To quantify the resulting improvement of real-time performance, we compare

TABLE VI
REAL-TIME PERFORMANCE COMPARISON OF DIFFERENT PROTECTION SCHEMES

Method	Group A		Group B		Group C	
	Steps	**RTC**	**Steps**	**RTC**	**Steps**	**RTC**
SpeechShield	**7**	**0.56**	**9**	**0.40**	**12**	**0.46**
SafeSpeech	110	6.27	125	3.37	160	5.61
AntiFake	150	6.93	160	3.72	165	3.80
SpeechShield [1*]	60	5.48	65	2.90	75	3.01
SpeechShield [2*]	100	6.47	110	3.57	110	3.04

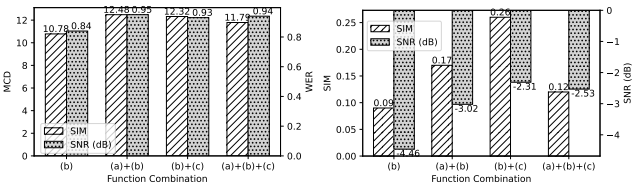
SpeechShield against two representative baselines and two degraded variants of SpeechShield: (1) SafeSpeech; (2) AntiFake; (3) SpeechShield without the meta-perturbation mechanism [1*]; (4) SpeechShield without the meta-perturbation and latent-space optimization mechanisms [2*]. Our audio samples are partitioned by duration into three groups to simulate different time-window constraints in real-time scenarios: (A) shorter than 5 seconds (average 3.4s), (B) between 5 and 8 seconds (average 7.8s), (C) longer than 8 seconds (average 9.8s). For each group, we vary the number of fine-tuning iterations. We record the average minimum iterations required to satisfy the strict protection thresholds (WER $\geq$ 90%, SIM $\leq$0.15, and SNR $\geq$–2.9dB when applicable) and measure the corresponding time needed for generation, and then compute the real-time coefficient (RTC).

As shown in Table VI, optimizing in the meta-perturbation latent space allows SpeechShield to achieve a 7x efficiency improvement over existing methods (up to 13x over SafeSpeech on Group C) while maintaining RTC$<$1. In Group A, SpeechShield requires only 1.9s to achieve 90.5% WER, significantly outperforming SafeSpeech (21.9s) and AntiFake (24.3s). For a 0.9s clip, 7 fine-tuning iterations yield 85% WER with RTC$\approx$1; relaxing the target to 75% WER requires only 5 iterations (0.65s). Without fine-tuning, we observe WER $\approx$ 69.6%, SIM $\approx$ 0.30, and SNR $\approx$ -3.09dB. The results imply that strong protection (WER$\geq$90%) is achievable within 2s (suitable for streaming), while strictly real-time scenarios can rely on 5 iterations of the meta-perturbation alone for moderate protection (WER$\geq$ 70%) with negligible delay.

We then apply Demucs v3 denoising to "meta-perturbation"-only samples, and the synthesized speech achieves 60.12% WER and 0.21 SIM. Although slightly reduced, protection remains effective (better than $<$20% baseline, as shown in Table I), and outperforms SOTA (Table IV). This indicates the meta-perturbation retains influence over speech features even without individualized fine-tuning. Practically, SpeechShield can be deployed as a virtual sound card plugin. Users can adjust fine-tuning steps and window length to balance latency and protection; we recommend minimizing both to satisfy real-time hardware constraints.

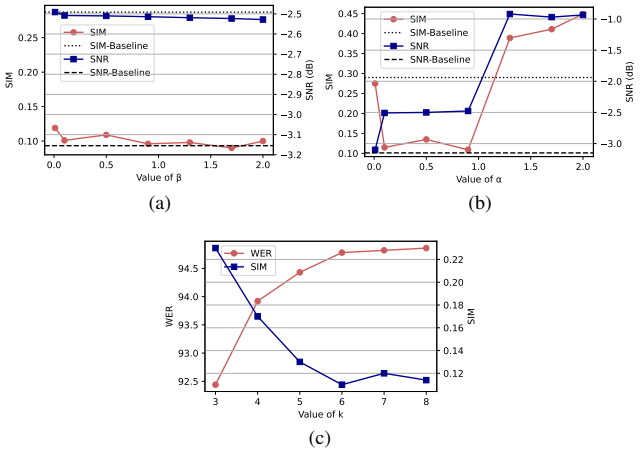
F. Ablation Study

Our voice protection is formulated as a multi-objective optimization problem. The optimization objectives and their



(a) (b)

Fig. 8. Ablation study results for different compositions of loss functions.



(a) (b) (c)

Fig. 9. Ablation results for different coefficients of $\mathcal{L}_r$ and $\mathcal{L}_g$, as well as k values.

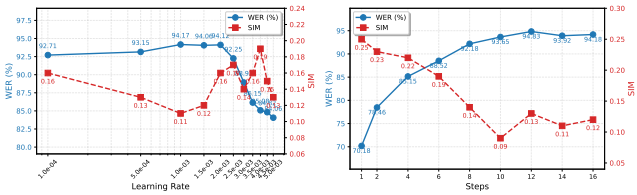
weights, as well as the number of harmonics, may all impact the protection performance. To investigate these factors, we conduct an ablation study on the LibriTTS dataset using the BERT-VITS2 surrogate TTS model.

Analysis of Loss Components. In Section III-C, we decompose our objective into three loss terms: (a) the efficacy loss $\mathcal{L}_e$, which drives the perturbation to be further effective; (b) the robustness loss $\mathcal{L}_r$, which makes the TTS model produce low-quality outputs, thereby undermining adversarial optimization; (c) the perceptual loss $\mathcal{L}_g$, which promotes human-perceived naturalness. We evaluate four combinations of these terms: (b), (a)+(b), (b)+(c), and (a)+(b)+(c), using the default weights from Section V-A. Since (b) is the key optimization objective for ensuring the robustness of perturbations, other combinations that exclude this objective (e.g., (a), (c) or (a+c)) are not considered. Figs. 8(a) and (b) show the WER and SIM for each configuration. The full combination, (a)+(b)+(c), makes the best overall trade-off, with MCD=11.79, WER=0.94, SIM=0.12, and SNR=-2.53dB. Using only (b) yields a high distortion and the noisiest SNR (MCD=14.08, SNR=-4.18dB), a comparable error WER=0.84 and the least similarity SIM=0.09. Since (b) only considers the distortion of acoustic features, auditory naturalness is greatly compromised. The pair, (a)+(b), yields slightly better MCD, WER, and introduces less audible noise (SNR = -3.02dB), perhaps because the introduction of the surrogate model makes the perturbation more closely coupled with the time segments where human voices are present. In summary, combining all three losses achieves the best balance between protection strength and perceptual quality.

Coefficient of Loss Functions. Different loss terms often operate on different scales and produce gradients of varying magnitudes, so naïvely summing them can cause one term to dominate the optimization. By tuning the scalar coefficient of each loss component, one can amplify or attenuate its influence and hence strike an appropriate balance among the different optimization objectives. We fix the coefficient of $\mathcal{L}_e$ at 1.0 and independently sweep the coefficients of $\mathcal{L}_r$ and $\mathcal{L}_g$ in [0.01,0.1,0.5,0.9,1.3,1.7,2.0], measuring adversarial strength via SIM and perceptual quality via SNR. Note that when varying the coefficient of $\mathcal{L}_r$, $\mathcal{L}_g$ is held at its default value of 0.1; conversely, when varying the coefficient of $\mathcal{L}_g$, $\mathcal{L}_r$ is fixed at 0.4, as mentioned in Section V-A. For comparison, we adopt the SIM and SNR results of SafeSpeech as the baseline. As shown in Fig. 9, when β increases from 0.01 to 0.4, SIM fluctuates around 1.1; when β is raised further into the range [0.4, 2.0], SIM stabilizes at approximately 0.09. At the same time, SNR slowly decreases as β grows. For γ, when it exceeds 1.0, SIM rises above 4.0, while SNR moves to around –1.0 dB. We observe that it is the relative balance among the loss weights that most strongly governs the tradeoff between protection and audio quality. In particular, once the perceptual weight γ exceeds 1.0, $\mathcal{L}_g$ begins to dominate over the efficacy loss $\mathcal{L}_e$. The pipeline then focuses on noise reduction and naturalness, which leads to a sudden increase in SIM and the corresponding decrease in the strength of adversarial perturbation.

Number of Harmonics k. In Eq. (3), the parameter k determines how many harmonics the perturbation covers, and the computations in Eqs. (3)-–(6) all depend on the bin indices b_k derived from k. We vary k from 3 to 8 while holding all other hyperparameters at their default values, and use WER and SIM to evaluate the protection performance under different k values. As shown in Fig. 9(c), when k is increased from 3 to 6, WER and SIM steadily increase and decrease, respectively, reaching approximately 94.8% and 0.11. However, when k increases beyond 6, WER only rises slightly and SIM fluctuates around 0.12. Overall, $k = 6$ is shown to be optimal because it covers the primary harmonic components that convey most of the speaker's timbral features, whereas additional, higher-order harmonics lie outside the key formant regions and contribute little to protection effectiveness.

Learning Rates and Fine-tune Steps. The learning rate decides the aggressiveness of weight updates following each data batch, while the number of fine-tuning steps (N) determines the total optimization iterations performed on a given utterance. Using utterances from speaker #3922 in the LibriTTS dataset, we evaluate fine-tuning learning rates (λ) across the range $[1e^{-4}, 5e^{-3}]$ at equal intervals, maintaining $N = 10$. To evaluate step count, we vary N between 1 and 20 with a fixed learning rate of $1e^{-3}$. As shown in Fig. 10, the protection performance varies with these parameters. When the learning rate exceeds $2e^{-3}$, the WER drops significantly, suggesting



(a) (b)

Fig. 10. Ablation study results for different learning rates and fine-tune steps.

that optimization fails to converge due to oscillations. The results indicate that a learning rate of $1e^{-3}$ provides the optimal balance of protection effectiveness under the given utterances. Furthermore, $N = 10$ offers the best trade-off between computational cost and protection, as both WER and SIM scores plateau for $N > 10$.

VI. Discussion

This section discusses the limitations of SpeechShield and our future work.

Multi-lingual Evaluation. Although our experimentation comprehensively evaluated speakers of different genders and accents, confirming SpeechShield's protection across these variations, real-world users may speak languages other than English (e.g., Spanish, Chinese, or Korean). Due to the lack of well-annotated datasets, we have not yet tested SpeechShield on other languages than English. In the future, we would like to evaluate SpeechShield with a broader range of languages and dialects to better understand its applicability in diverse linguistic contexts.

Support for Multi-Speaker Audio. Our design and experimentation assume that each input audio sample contains only a single speaker's voice. However, in scenarios such as online conversations, voice recordings may include the victim's speech mixed with that of unrelated speakers. Under these conditions, our optimization could unintentionally apply personalized perturbations to all voices in the mix rather than exclusively to the target speaker. Thus, we plan to integrate speech-separation techniques so as to generate and apply perturbations only to the victim's vocal track.

Enhancing Long-Term Effectiveness. Our optimization leverages adversarial loss functions to prevent TTS models from synthesizing speech with clear, accurate timbre, while also resisting existing adversarial training. However, as AI advances, new generations of TTS models and attacks may weaken existing defenses. To stay ahead, we plan to integrate insights from unlearning or embedding-based attacks to optimize our strategies, ensuring that SpeechShield remains effective and resilient.

VII. Related Work

Voice Synthesis Models. TTS models extract a speaker's vocal characteristics and then generate spoken output by conditioning on phoneme features to match a given text. Early sequence-to-sequence approaches [47], [48] use encoder–decoder attention to predict mel-spectrograms from the text, achieving high naturalness but at the cost of slow, autoregressive inference. Subsequent non-autoregressive systems [49], [14], [23] improve inference speed by predicting spectrograms in parallel, albeit sometimes with prosody trade-offs. More recently, diffusion probabilistic models [50], [51] have made further gains in output quality, rivaling top conventional TTS in naturalness, while offering an adjustable trade-off between generation speed and fidelity. In parallel, large-scale self-supervised and pre-trained models [12], [52] have treated TTS as a conditional language modeling problem using discrete audio codes, which enables prompt-based zero-shot voice synthesis.

Anti-Voice Synthesis and Voice Cloning. To counter unauthorized voice cloning and speech deepfakes, prior research has explored both proactive defenses and detection mechanisms. For proactive defenses, adversarial noise can significantly disrupt zero-shot voice conversion models while remaining perceptual-friendly to human listeners [53], [3], [2]. Moreover, signal processing techniques (e.g., filtering or formant shifting via vocal tract length normalization) and noise addition have also been used to mask speaker-identifying features [54], [55]. The detection strategy is to embed a hidden "timbre watermark" in a speaker's voice recording that remains through synthesis. The embedded watermark can be used as part of the voice features and will be learned by TTS models, thereby allowing synthesized copies to be detected post hoc [5], [56], [57].

VIII. Conclusion

In this paper, we have presented the first real-time and robust protection framework, SpeechShield, against voice synthesis attacks. We have carefully analyzed the critical acoustic components of the speech signal and devised two main optimizations: harmonic energy and spectral flatness. By strategically injecting perturbations into low-order harmonics and the gaps between harmonic peaks, SpeechShield effectively prevents TTS models from synthesizing speeches with intelligible voice content. SpeechShield better withstands various adversarial data processing and training attacks, demonstrating stronger robustness than SOTA methods. Furthermore, by projecting the perturbation optimization process into a compact latent space and using a meta-perturbation strategy, SpeechShield can generate an effective, perceptually-friendly perturbation under the latency constraints of real-time applications, making it applicable for time-sensitive applications. Our extensive experiments corroborate that SpeechShield outperforms SOTA baselines in terms of protection effectiveness, robustness, and timeliness efficiency.

Acknowledgments

We would like to thank the anonymous reviewers for their constructive and insightful feedback. This work was supported in part by the National Natural Science Foundation of China under Grant Number 62472302 and the U.S. National Science Foundation (NSF) under Grant Numbers CNS-2245223 and CNS-2317829.

REFERENCES

[1] Y. A. Li, C. Han, and N. Mesgarani, "Styletts: A style-based generative model for natural and diverse text-to-speech synthesis," *IEEE Journal of Selected Topics in Signal Processing*, 2025.

[2] Z. Yu, S. Zhai, and N. Zhang, "Antifake: Using adversarial audio to prevent unauthorized speech synthesis," in *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2023.

[3] Z. Zhang, D. Wang, Q. Yang, P. Huang, J. Pu, Y. Cao, K. Ye, J. Hao, and Y. Yang, "Safespeech: Robust and universal voice protection against malicious speech synthesis," in *Proceedings of the USENIX Security Symposium (USENIX Security)*, 2025.

[4] X. Zhang, X. Sun, X. Sun, W. Sun *et al.*, "Robust reversible audio watermarking scheme for telemedicine and privacy protection," *Computers, Materials & Continua*, vol. 71, no. 2, 2022.

[5] C. Liu, J. Zhang, T. Zhang, X. Yang, W. Zhang, and N. Yu, "Detecting voice cloning attacks via timbre watermarking," in *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2024.

[6] W. Zong, Y.-W. Chow, W. Susilo, J. Baek, and S. Camtepe, "Audiomarknet: Audio watermarking for deepfake speech detection," in *Proceedings of the USENIX Security Symposium (USENIX Security)*, 2025.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proceedings of the Springer Medical Image Computing and Computer-assisted Intervention (MICCAI)*, 2015.

[9] G. Chen, Y. Zhang, F. Song, T. Wang, X. Du, and Y. Liu, "Songbsab: A dual prevention approach against singing voice conversion based illegal song covers," in *Proceedings of the Annual Network and Distributed System Security Symposium (NDSS)*, 2025.

[10] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry, "Adversarial examples are not bugs, they are features," *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[11] B. Li, Y. Wei, Y. Fu, Z. Wang, Y. Li, J. Zhang, R. Wang, and T. Zhang, "Towards reliable verification of unauthorized data usage in personalized text-to-image diffusion models," in *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2025.

[12] S. Chen, S. Liu, L. Zhou, Y. Liu, X. Tan, J. Li, S. Zhao, Y. Qian, and F. Wei, "Vall-e 2: Neural codec language models are human parity zero-shot text to speech synthesizers," *arXiv preprint arXiv:2406.05370*, 2024.

[13] I. Elias, H. Zen, J. Shen, Y. Zhang, Y. Jia, R. Skerry-Ryan, and Y. Wu, "Parallel Tacotron 2: A non-autoregressive neural TTS model with differentiable duration modeling," *arXiv preprint arXiv:2103.14574*, 2021.

[14] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech 2: Fast and high-quality end-to-end text to speech," *arXiv preprint arXiv:2006.04558*, 2020.

[15] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, K. Kavukcuoglu *et al.*, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[16] J. Kong, J. Kim, and J. Bae, "Hifi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," in *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[17] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.

[18] R. J. Baken, *Clinical measurement of speech and voice*, 2nd ed. Taylor & Francis Ltd., 2000.

[19] L. Fowl, M. Goldblum, P.-y. Chiang, J. Geiping, W. Czaja, and T. Goldstein, "Adversarial examples make strong poisons," in *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[20] A. Défossez, "Hybrid spectrogram and waveform source separation," *arXiv preprint arXiv:2111.03600*, 2021.

[21] A. De Cheveigné and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," *The Acoustical Society of America*, vol. 111, no. 4, pp. 1917–1930, 2002.

[22] F. Jacobsen and P. M. Juhl, *Fundamentals of general linear acoustics*. John Wiley & Sons, 2013.

[23] J. Kim, S. Kim, J. Kong, and S. Yoon, "Glow-TTS: A generative flow for text-to-speech via monotonic alignment search," *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[24] Y. Wang, H. Guo, G. Wang, B. Chen, and Q. Yan, "Vsmask: Defending against voice synthesis attack via real-time predictive perturbation," in *Proceedings of the ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)*, 2023.

[25] M. Tagliasacchi, Y. Li, K. Misiunas, and D. Roblek, "SEANet: A multi-modal speech enhancement network," *arXiv preprint arXiv:2009.02095*, 2020.

[26] H. Zhang, X. Zhang, and G. Gao, "Training supervised speech separation system to improve stoi and pesq directly," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[27] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[28] A. Nichol, J. Achiam, and J. Schulman, "On first-order meta-learning algorithms," *arXiv preprint arXiv:1803.02999*, 2018.

[29] S. Chen, G. Yuan, X. Cheng, Y. Gong, M. Qin, Y. Wang, and X. Huang, "Self-ensemble protection: Training checkpoints are good data protectors," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

[30] H. Huang, X. Ma, S. M. Erfani, J. Bailey, and Y. Wang, "Unlearnable examples: Making personal data unexploitable," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[31] "LibriTTS Dataset," https://www.openslr.org/60/, accessed: 2025-05.

[32] "CMU Arctic Speech Databases," http://www.festvox.org/cmu_arctic/, accessed: 2025-05.

[33] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2023.

[34] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *arXiv preprint arXiv:2005.07143*, 2020.

[35] R. Kubichek, "Mel-cepstral distance measure for objective speech quality assessment," in *Proceedings of the IEEE Pacific Rim Conference on Communications Computers and Signal Processing (PacRIM)*, 1993.

[36] C.-y. Huang, Y. Y. Lin, H.-y. Lee, and L.-s. Lee, "Defending your voice: Adversarial attack on voice conversion," in *IEEE Spoken Language Technology Workshop (SLT)*, 2021.

[37] "BERT-VITS2," https://github.com/fishaudio/Bert-VITS2, accessed: 2025-05.

[38] "MB-iSTFT-VITS," https://github.com/MasayaKawamura/MB-iSTFT-VITS, accessed: 2025-05.

[39] "VITS," https://github.com/jaywalnut310/vits, accessed: 2025-05.

[40] "Glow-TTS," https://github.com/jaywalnut310/glow-tts, accessed: 2025-05.

[41] "StyleTTS 2," https://github.com/yl4579/StyleTTS2, accessed: 2025-05.

[42] "Demucs v3," https://github.com/facebookresearch/demucs, accessed: 2025-05.

[43] "AudioPure," https://github.com/cychomatica/AudioPure, accessed: 2025-05.

[44] "MDX23C," https://github.com/ZFTurbo/MVSEP-MDX23-music-separation-model, accessed: 2025-05.

[45] "BS-Roformer," https://github.com/lucidrains/BS-RoFormer, accessed: 2025-05.

[46] W. Fan, K. Chen, C. Liu, W. Zhang, and N. Yu, "De-antifake: Rethinking the protective perturbations against voice cloning attacks," *arXiv preprint arXiv:2507.02606*, 2025.

[47] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," in *Proceedings of the IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 2018.

[48] J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastner, A. Courville, and Y. Bengio, "Char2wav: End-to-end speech synthesis," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

[49] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech: Fast, robust and controllable text to speech," in *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[50] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. Kudinov, "Grad-tts: A diffusion probabilistic model for text-to-speech," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2021.
[51] R. Huang, Z. Zhao, H. Liu, J. Liu, C. Cui, and Y. Ren, "Prodiff: Progressive fast diffusion model for high-quality text-to-speech," in *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, 2022.
[52] B. Han, L. Zhou, S. Liu, S. Chen, L. Meng, Y. Qian, Y. Liu, S. Zhao, J. Li, and F. Wei, "Vall-e r: Robust and efficient zero-shot text-to-speech synthesis via monotonic alignment," *arXiv preprint arXiv:2406.07855*, 2024.
[53] J. Li, D. Ye, L. Tang, C. Chen, and S. Hu, "Voice guard: Protecting voice privacy with strong and imperceptible adversarial perturbation in the time domain," in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2023.
[54] G. P. Prajapati, D. K. Singh, P. P. Amin, and H. A. Patil, "Voice privacy through x-vector and cyclegan-based anonymization." in *Proceedings of the Interspeech Conference*, 2021.
[55] J. Deng, F. Teng, Y. Chen, X. Chen, Z. Wang, and W. Xu, "V-cloak: Intelligibility-, naturalness-& timbre-preserving real-time voice anonymization," in *Proceedings of the USENIX Security Symposium (USENIX Security)*, 2023.
[56] A. A. AlSabhany, A. H. Ali, and M. Alsaadi, "A lightweight fragile audio watermarking method using nested hashes for self-authentication and tamper-proof," *Multimedia Tools and Applications*, pp. 1–15, 2024.
[57] I. Natgunanathan, P. Praitheeshan, L. Gao, Y. Xiang, and L. Pan, "Blockchain-based audio watermarking technique for multimedia copyright protection in distribution networks," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 18, no. 3, pp. 1–23, 2022.

ETHICAL CONSIDERATIONS

Dual-use risks. While adversarial research carries inherent dual-use risks, our contributions are strictly defensive. We focus on enhancing robustness against attacks rather than facilitating them. By documenting these defense mechanisms, we aim to stay ahead of malicious exploitation rather than providing a blueprint for it. **Data usage.** All experiments utilized standard, publicly available datasets (LibriTTS, CMU Arctic). No private, sensitive, or personally identifiable recordings were collected, and no human subjects were recruited. All work complies with the original data licensing and established research norms. **Responsible disclosure and societal benefit.** Our findings highlight critical vulnerabilities in voice-cloning defenses. To maximize societal benefit while minimizing risk, we provide detailed methodological descriptions to support reproducibility in defense research but refrain from releasing tools that could be directly weaponized for large-scale attacks.

OPEN SCIENCE

We release our artifact to the anonymous repository, see link https://anonymous.4open.science/r/SpeechShield-Artifact-8851/.