SafeSpeech: Robust and Universal Voice Protection Against Malicious Speech Synthesis

Zhisheng Zhang[1], Derui Wang[2] ✉, Qianyi Yang[1], Pengyang Huang[1], Junhan Pu[1], Yuxin Cao[3], Kai Ye[4], Jie Hao[1] ✉, and Yixian Yang[1]

[1]*Beijing University of Posts and Telecommunications* [2]*CSIRO's Data61*

[3]*National University of Singapore* [4]*The University of Hong Kong*

Abstract

Speech synthesis technology has brought great convenience, while the widespread usage of realistic deepfake audio has triggered hazards. Malicious adversaries may unauthorizedly collect victims' speeches and clone a similar voice for illegal exploitation (*e.g.*, telecom fraud). However, the existing defense methods cannot effectively prevent deepfake exploitation and are vulnerable to robust training techniques. Therefore, a more effective and robust data protection method is urgently needed. In response, we propose a defensive framework, ***SafeSpeech***, which protects the users' audio before uploading by embedding imperceptible perturbations on original speeches to prevent high-quality synthetic speech. In SafeSpeech, we devise a robust and universal proactive protection technique, **S**peech **PE**rturbative **C**oncealment (**SPEC**), that leverages a surrogate model to generate universally applicable perturbation for generative synthetic models. Moreover, we optimize the human perception of embedded perturbation in terms of time and frequency domains. To evaluate our method comprehensively, we conduct extensive experiments across advanced models and datasets, both subjectively and objectively. Our experimental results demonstrate that SafeSpeech achieves state-of-the-art (SOTA) voice protection effectiveness and transferability and is highly robust against advanced adaptive adversaries. Moreover, SafeSpeech has real-time capability in real-world tests. The source code is available at https://github.com/wxzyd123/SafeSpeech.

1 Introduction

In recent years, the rapid growth of generative artificial intelligence (AI) [4] has drawn broad social attention. People are amazed by the excellent capabilities of AI, which benefits from the continuous progress in deep neural networks (DNNs). In speech synthesis, or voice cloning, models trained on large-scale speech corpus can now generate highly realistic audio [27–29]. Through fine-tuning pre-trained models, the latest one needs only a few minutes of speech samples to synthesize high-quality speeches with realistic timbre, rhythm, and phonemes. Although early voice cloning technology is mostly used for positive purposes, such as cloning deceased lovers' voices to provide comfort, there have recently been cases of this tool being abused for illegal activities, *e.g.*, Figure 1. Moreover, criminals used deepfake speech to pose as a German boss and tricked a British subsidiary head into transferring $243,000 [54]. Tackling deepfake speech is vital for the integrity and security of voice-based systems in daily life.

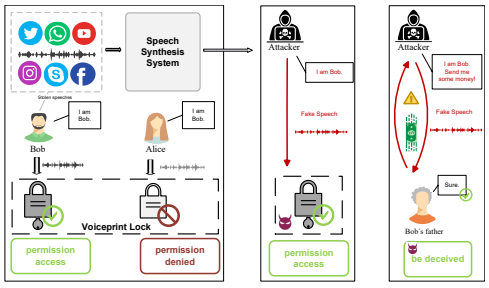


Figure 1: Speech synthesis hazards in real-world scenarios, *e.g.*, the attacker utilizes Bob's public voices with TTS tools to bypass the voiceprint lock and achieve telecom fraud.

Existing Defenses and Limitations. To counter the threat of deepfake speech, existing voice protection methods like AntiFake [68], VSMask [61], and AttackVC [19], focus on leveraging adversarial examples to make the synthetic samples do not resemble the original speaker in terms of timbre, preventing zero-shot speech synthesis (or voice conversion).

However, although previous methods have certain effects, they also have serious limitations: (1) *Protection Scenarios*. Current voice protection techniques focus on zero-shot scenarios, *i.e.*, employing one reference audio to clone voice during the inference stage. However, in addition to zero-shot scenarios, adversaries may also fine-tune models, which poses a more severe challenge for two reasons. Firstly, many models do not support zero-shot systhesis [27–29, 62], and fine-

✉ Corresponding authors: derek.wang@data61.csiro.au, haojie@bupt.edu.cn.

tuning can achieve better quality. Secondly, previous methods based on adversarial examples cannot withstand fine-tuning techniques. (2) *Synthesis Quality Prevention*. Previous voice protection methods on adversarial examples can generate dissimilar but high-quality deepfake speeches, which means that these speeches can still be utilized. However, we aim to make the synthesized audio significantly low-quality and cannot be utilized to address the deepfake issue fundamentally, *i.e.*, *synthesis quality prevention*. High-quality deepfake speeches pose security risks. First, adversaries can conduct large-scale searches for new victims with similar voices. Second, synthetic audio might still be utilized maliciously, such as voice assistants in telecom fraud, to conceal the true identity.

Motivation. Many regions have introduced regulations on generative AI governance and data protection, like California's Consumer Privacy Act (CCPA) [34], making voice privacy protection urgent. We conduct this work for two motivations. Firstly, we aim for broader voice protection, covering training time and deepfake audio quality. Fine-tuning-based speech synthesis is crucial as it can cover more TTS models and produce higher-quality audio. Secondly, large language models (LLMs) [4, 58] have developed continuously. LLMs can generate high-quality human-like text for TTS models, promoting realistic deepfake speech production. Synthesizers incorporating LLMs (*e.g.*, BERT-VITS2 [1] and GPT-SoVITS [2]) also need consideration.

Technical Challenges. In addressing the aforementioned issues, we have to overcome these challenges: (1) *Effectiveness and Transferability*. We need to design voice protection that is effective against fine-tuning. Additionally, the algorithm should possess strong transferability across various TTS models. (2) *Modal Selection*. The input of TTS models is multimodal, such as waveform, spectrogram, and corresponding text. It is crucial to decide the most sensible modality of the anti-learning perturbations. (3) *Robustness.* Previous data protection methods [20, 68] are vulnerable to adaptive training [63]. Therefore, our method should be robust against advanced adaptive adversaries for real-world applications. (4) *Imperceptibility.* The embedded perturbation should be imperceptible or align with human perception, necessitating a design optimization method for noise incorporation that ensures human acceptance or harmlessness. Overall, successful voice protection should satisfy the prevention of synthetic speech intelligibility (*i.e.*, synthesis quality) and speaker timbre similarity against training-stage voice cloning.

Our Response Strategies. In response to these challenges, we introduce ***SafeSpeech***, a framework to safeguard data by embedding specifically designed perturbation while preserving text consistency before audio uploading. To effectively protect voice at training time and enhance transferability, we propose pivotal objective optimization with less computational time based on a surrogate model. Additionally, to achieve further protection, we introduce the Speech PErturbative Concealment (SPEC) techniques based on Kullback-Leibler divergence, which better conceals speech information. These approaches lead to voice protection in terms of speech quality and timbre similarity. To optimize the audibility of embedded noise, ℓ_p norm may not fully adapt to human [68] and we devise perceptual optimization functions to reduce human audibility. The safeguarded audio by our proposed SafeSpeech ensures that the synthesized audio is not similar and undermines the speeches' usability, thus providing a more effective and robust defense against various adaptive attackers utilizing a novel voice protection method.

Experiments and Evaluation. To validate the effectiveness of SafeSpeech, we conduct experiments on current SOTA TTS models and well-known datasets across various metrics. Our extensive experiments demonstrate that SafeSpeech achieves the SOTA protection effect against train/test-time voice cloning. SafeSpeech is also highly robust facing perturbation removal, data augmentation, model (or data) recovery, and adaptive robust training, *etc*. Moreover, the physical-world test demonstrates that SafeSpeech possesses high robustness and real-time capabilities with continuous on-site voice protection. In the user study, we evaluate the human perception and most participants believe synthetic speeches after protection cannot deceive them. Compared to previous methods [19, 61, 68], SafeSpeech has stronger effectiveness and robustness, preventing fine-tuning-based speech synthesis.

Novelty and Contributions. In this paper, we achieve innovation in three levels: (1) *Algorithm*. We propose the pivotal objective optimization and the SPEC technique, which can achieve better effect, transferability, robustness, and efficiency. (2) *Scenario*. Compared to previous literature [19, 61, 68], we consider the fine-tuning strategies, presenting a more protective scenario than zero-shot voice cloning. (3) *Application*. By utilizing a lightweight surrogate model and simplified objective, we can achieve real-time protection in real-world applications. The CCPA emphasizes personal data privacy and regulates business data handling. Our SafeSpeech prevents unauthorized and malicious voice exploitation. Our main contributions can be summarized as follows:

- We propose ***SafeSpeech*** which for the first time protects our voice at training time in our best knowledge by embedding imperceptible perturbation against unauthorized exploitation and malicious speech synthesis.
- We devise a robust and universal perturbative technique named Speech PErturbative Concealment against malicious speech synthesis. For noise imperceptibility, we introduce a hybrid perceptual function, combining STOI and STFT loss, to optimize human perception and reduce inaudibility in terms of time and frequency domains.
- We comprehensively evaluate SafeSpeech across ten SOTA models and two datasets during training and testing phases. The SafeSpeech is robust against adaptive adversaries.
- SafeSpeech can achieve real-time protection in our real-world test and takes only 10.606 seconds to generate speaker-specific perturbation with continuous protection.

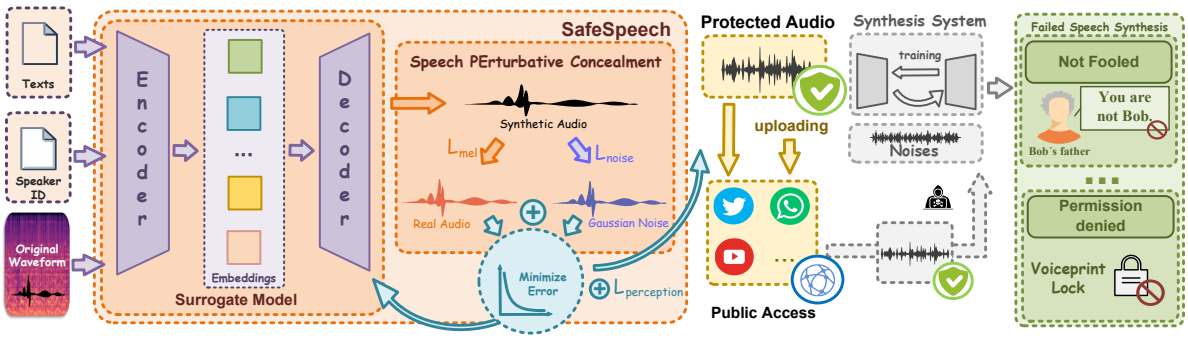


Figure 2: The SafeSpeech safeguards voice by constructing a surrogate TTS model that minimizes the designed objectives ($\mathcal{L}_{mel}$ and $\mathcal{L}_{noise}$ with perception constraint $\mathcal{L}_{perception}$ detailed in Section 4). Despite attackers fine-tuning advanced TTS models from social platforms, they cannot produce high-quality synthetic speech to circumvent voiceprint locks or deceive victims' families.

2 Preliminaries

Mainstream speech synthesis utilizes a DNN-based model to input signals with timbre and audio features, and these models usually consist of an encoder and decoder architecture (*e.g.*, Figure 2). Compared to traditional rule-based synthesis methods [43], current TTS models [1, 27, 29] can achieve a better voice cloning effect with a few samples. In this context, we explore privacy preservation strategies for TTS synthesis.

Voice Anti-Cloning. AntiFake [68], VSMask [61], and AttackVC [19] are three voice protection methods based on adversarial examples ensuring zero-shot TTS models cannot synthesize voiceprint-similar speech. Our SafeSpeech, in contrast, goes beyond mere voiceprint similarity and inference stage protection, as it actively prevents the usability of deepfake speech. SafeSpeech considers the protection of the timbre feature and synthesis quality *based on unlearnable examples, a training stage data protection technique*.

Data Poisoning. Our method can be regarded as a special type of clean-label and triggerless data poisoning, while the task purposes are significantly distinct. Data poisoning attack aims to degrade the model's performance on clean samples by modifying training samples. Previous data poisoning attacks [41, 65] have focused on identifying the most influential samples to affect model learning and modifying these training samples (*e.g.*, changing labels [56] or embedding large perturbations [65]). The primary aim of poisoning attacks is to degrade the overall performance of the model after poisoning a portion of the data resulting in the inability to use the authorized data normally, while in the protective scenario discussed in this paper, we aim to protect audio data rather than poisoning the model to degrade its performance. In other words, the perturbations added by SafeSpeech affect the protected samples without affecting the unprotected ones. This point will be discussed in Section 7.2.3 by experiments.

Unlearnable Examples. In classification tasks, let a DNN-based classifier be f, which accepts input data x with corresponding label y. We regard the clean training and testing datasets as D_c and D_t, respectively. The creation of unlearnable examples is facilitated by a perturbation optimizer that treats D_c as the input and produces an unlearnable dataset D_u by embedding a perturbation δ to samples in D_c. When training on D_c, the model f demonstrates excellent performance on D_t but suffers from poor performance when training on D_u. The perturbation generation relies on a bi-level structure [20] which optimizes both the perturbation and the parameter:

$$\arg\min_{\theta} \mathbb{E}_{x,y}[\min_{\delta} \mathcal{L}(f_\theta(x+\delta), y)], \tag{1}$$

where f_θ is a classifier with the trainable parameter θ.

Research [16, 20] indicates that minimizing errors in the training objectives can initially disrupt the training process. However, to achieve effective and robust voice protection, more advanced techniques must be designed.

3 Threat Models

In the threat model, we introduce *a priori* knowledge, capability, and limitations of the adversary, defender, and system.

3.1 Adversary Capability

We assume that the adversary is the third-party entity that can utilize the current most advanced TTS models to achieve successful voice cloning of the victim on unprotected data. To simulate experienced adversaries in the real world, we consider three of their capabilities:

Capability of Data Access. The development of the Internet has exposed more data to the public with potential threats. Adversaries can directly download the uploaded audio of the victim from public media on the Internet (*e.g.*, YouTube, Facebook) using web crawler technology or some permission bypass mechanisms. They cannot access the original unprotected audio if obtained speeches are protected.

Capability of Customized and Robust Training. Adversaries can leverage various advanced speech synthesis models.

Attackers can achieve speech synthesis to bypass speaker verification or human perception. At the same time, we consider a stronger adversary, that is, the adversary can detect the abnormal perturbations embedded for protection and employs the most advanced defensive data augmentation and robustness training, such as perturbation removal, adversarial training, specific data poisoning defensive methods, and speech transformation, to seek high-quality speech synthesis.

Capability of Model Recovery. We assume that the adversary is an experienced model trainer. If the speech synthesis does not achieve the desired effect after fine-tuning the TTS model utilizing the acquired audio, then he may realize that the model has been poisoned and restore it to the initial state against the protection strength detailed in Section 7.2.1.

3.2 Defender Description

Defender Limitations. To more faithfully replicate real-world scenarios, we restrict the defender to only having access to the synthesized audio from the model's output and is unaware of the model training method of the attacker. The defender can only introduce perturbation to the original audio. Additionally, we restrict that the generation of noise does not depend on *a priori* knowledge for users' application, *e.g.*, the utilized model for speech synthesis, and the protection of the speaker's samples can poison all or only part of the samples. Moreover, we assume the uploading condition when the protected audio can defend potential and the most advanced data augmentation and robust training, *etc.*, strategies for confidential exploitation in the real world after uploading the perturbative protected samples.

System Capabilities. System, *i.e.*, SafeSpeech in this paper, needs to effectively generate perturbation for the audio that users want to protect before uploading. For the aim of voice protection, SafeSpeech should generate perturbation without affecting the use of unprotected audio. Moreover, the generated perturbation should be imperceptible or at least perceptually acceptable. Audio protected by SafeSpeech should achieve timbre and synthesis quality protection, while also ensuring transferability across different models. In conclusion, we design our protective system for the two aims:

- **System Aim 1 (SA1): Timbre Protection.** Our goal is to achieve a state where fine-tuning on protected audio cannot synthesize audio that resembles the target victim.
- **System Aim 2 (SA2): Quality Protection.** The "synthesis" quality protection represents the synthetic speech is low-quality and cannot be utilized normally in daily life.

Previous adversarial-examples-based voice protection [19, 61, 68] can only achieve SA1 in zero-shot scenarios. However, to fundamentally prevent deepfake audio, the defender needs to meet SA1 and SA2 in zero-shot and fine-tuning scenarios.

4 SafeSpeech Methodology

In response to speech synthesis defense at training time, we design SafeSpeech to achieve voice anti-cloning. Figure 2 presents the workflow of SafeSpeech and the attacker's malicious action. For data protection, we introduce the optimization objective and propose the proactive defense mechanisms of the pivotal function and SPEC. While, for perception optimization, we introduce the perception metrics, *i.e.*, STFT, and STOI, to better the human perception of protected samples.

Problem Formulation. As we introduced in Section 1, we aim to prevent high-quality deepfake audio generation and propose a universal and robust perturbative voice protection method. Based on this, we design to solve an error-minimizing problem including effect and perception. We express the objectives of SafeSpeech by the following formula:

$$\begin{aligned} \arg\min_{\delta} \; & \mathcal{L}(G(x+\delta),x) + \alpha P(x+\delta), \\ s.t. \; & H(G(x+\delta)) \neq H(G(x)), \\ & SV(G(x+\delta)) \neq SV(G(x)), \\ & H(x+\delta) \approx H(x), \end{aligned} \tag{2}$$

where $G(\cdot)$, $SV(\cdot)$ are a speech synthesizer and speaker verification system respectively, $P(\cdot)$ is the auditory function, and $H(\cdot)$ is the human perception according to input audio. $\mathcal{L}(\cdot)$ is the objective, x is an input audio, α is a weight coefficient, and δ is the perturbation bounded by ℓ_p norm as $||\delta||_p \leq$ radius ε for the limitation of human perception.

4.1 Data Protection

To effectively mitigate the unauthorized speech synthesis in real-world scenarios, the generation process of protected audio should be devoid of reliance on *a priori* knowledge. This is crucial as we cannot predict the training strategies of the adaptive attackers. In the design of SafeSpeech, we aim to produce an effective, robust, and universal perturbation preventing training-time speech synthesis across different models.

Unlearnable Audio. Previous voice protection based on adversarial examples cannot be effective during the training stage, which is an unavoidable scenario. We aim to solve this based on training-stage data protection [20] and introduce an *error-minimizing* (EM) noise. The EM problem reduces the error of the model simulating the normal training process by perturbation so that there is "nothing" to learn when training on the safeguarded dataset as we introduced in Eq. (1).

It is significant to decide the optimization objective, because TTS models often engage in multi-task learning with multimodal inputs, like audio accompanied with relevant text guidance. Under the assumption of the defender, we can only modify the user's original audio while preserving the integrity of the text input. Generative speech synthesis models typically learn from the input data and generate outputs with

similar distributions. They may also incorporate discriminators or speaker encoders to improve the high-quality audio generation. However, considering the various architectures of different models, we focus on optimizing the generator, as it is most related to output information and audio quality.

By optimizing the objective function using perturbation, the error of the model has been greatly reduced and it is possible to make audio unlearnable for the model to think that there is "nothing" to learn. The objective function of the generator g from a TTS model can be expressed by following the single-level loop with multi-modal inputs:

$$\mathcal{L}_{TTS} = \sum_{i=0}^{k} \mathcal{L}_i \left[g(\texttt{text}, \texttt{spec}(x+\delta)), x, \theta \right], \tag{3}$$

where x represents the raw waveform, and $\text{spec}(\cdot)$ computes the linear spectrogram from inputs, and θ is the parameter.

Therefore, the core solution is to decide $\mathcal{L}_i$.

Pivotal Objective Selection. When applying noise to optimize the multi-task learning problem, directly using the objectives of TTS models yields unsatisfactory perturbative results (Section 6.4). Moreover, this is highly inefficient, requiring specifying a unique optimization approach for each TTS model. In this part, we will first illustrate why direct optimization is ineffective through examples, and then proceed to carefully analyze the features of TTS models and propose our pivotal objective function and its principles to be satisfied.

Different TTS models own different objective functions and components. For instance, BERT-VITS2 [1] comprises four components, with the generator containing eight objective functions, and the optimization function can be expressed as:

$$\begin{aligned} \mathcal{L}_G = \mathcal{L}_{recon} + \mathcal{L}_{kl} + \mathcal{L}_{dur} + \mathcal{L}_{adv}(G) \\ + \mathcal{L}_{fm}(G) + \mathcal{L}_{DurD} + \mathcal{L}_{score} + \mathcal{L}_{encoder}, \end{aligned} \tag{4}$$

where $\mathcal{L}_{recon}$ denotes the reconstruction loss between ground truth and generated speech. $\mathcal{L}_{kl}$ and $\mathcal{L}_{dur}$ represent the KL divergence loss and duration loss. $\mathcal{L}_{adv}$ and $\mathcal{L}_{fm}$ are the adversarial training loss and feature-matching loss of the generator. $\mathcal{L}_{DurD}$ is the duration discriminator loss, $\mathcal{L}_{score}$ computes the similarity score of embeddings from the generated and real audio, and $\mathcal{L}_{encoder}$ represents the encoder loss.

Among these, the duration loss $\mathcal{L}_{dur}$ only depends on the input text and cannot be optimized via perturbation. Furthermore, VITS [29] has five objective functions that are different from BERT-VITS2. Consequently, directly applying the eight-loss setup from BERT-VITS2 for perturbation may not yield a universally applicable approach in VITS or other models.

Moreover, through a thorough analysis of the objective function, we realize that the optimization effect is closely related to the multi-modal input characteristics of the model. We can only interfere with audio waveform, so that objective functions unrelated to audio can not be effectively affected. Furthermore, due to the various structures and optimization objectives of different models, it is advisable to devise a *universal* function adapting different generative TTS models for the perturbation universality.

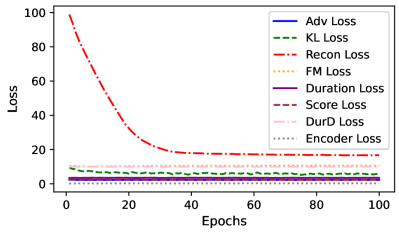


Figure 3: The convergence speed comparison of different objective functions when optimizing by perturbation.

The selection of an optimization objective is crucial given our lack of knowledge about the attacker's training model and structure. So, we should stick to the following listed principles of the pivotal function selection:

(a) The objective function can be optimized by perturbation;
(b) To ensure that universal perturbation is independent of *a priori* knowledge, the objective function should be universal across various TTS models;
(c) The function should be easily optimized through perturbation, such as achieving a rapid rate of convergence or containing relatively rich information entropy.

For example, in Eq. (4), the $\mathcal{L}_{dur}$ is not related to the waveform x violating the principle (a). These three principles must be considered in data protection guiding the transferability of the perturbation. When designing SafeSpeech, it is crucial to ensure that the users can perform perturbations regardless of the model they are using. This poses a challenging question:

> *Is it possible to devise a perturbative method that is universally applicable for all generative TTS models?*

In the scenario of this paper, a universal perturbation method implies identifying an optimization target that is consistent across different models. Upon further reflection, we recognize that generative TTS models typically aim to output audio or spectrograms that follow a distribution similar to the input [27–29, 35, 36, 48, 50, 51, 55], with training involving fitting these distributions to optimize the generator. Therefore, we propose to measure the *distance between the model's output and the input waveform as our pivotal optimization target*. Drawing from common TTS objective functions, we select the ℓ_1 distance to compute the similarity between the mel-spectrograms of the synthesized audio $\hat{x}$ and real input x. For end-to-end TTS, we first compute the mel-spectrograms for both the output and the input audio. If the model outputs spectrograms, we can then directly optimize them. This mel function can be formulated as:

$$\mathcal{L}_{mel} = ||x_{mel} - \hat{x}_{mel}||_1. \tag{5}$$

The reason for choosing the function $\mathcal{L}_{mel}$ as our pivotal

objective lies in the fact that generative TTS models serve the synthetic speeches $\hat{x}$ as the models' outputs. Crucially, we possess the capability to calculate the ℓ_1 distance between these outputs and real speeches, a computation that remains agnostic to the models' complex architecture. The `mel` optimization function $\mathcal{L}_{mel}$ also satisfies the principles (a) and (b). In the following, we analyze the selection from the perspective of convergence speed of principle (c). We use 10 samples for a batch and optimize them for 100 epochs on BERT-VITS2. The convergence speed is presented in Figure 3. We can observe that the `recon` function $\mathcal{L}_{mel}$ has the fastest convergence speed, starting from the highest initial value of 98.8 and eventually dropping to 16.7, and approaches convergence at the 40th epoch. In contrast, other objective functions have slow convergence speeds with hardly any noticeable decrease, so they are hard to optimize via perturbation. Therefore, we target the objective function with the fastest convergence speed and easiest optimization as a part of the data protection.

Compared with vanilla objective optimization, *i.e.*, initial unlearnable examples [20], selecting a generalized and fastest-converging function from multi-objective functions for optimization can enhance the effectiveness of protection and reduce time costs. This is because, when using perturbation to optimize the multi-task objectives, the values in the gradient space are influenced by the optimization directions of multiple functions, making it difficult for the loss to converge. In contrast, choosing $\mathcal{L}_{mel}$ allows the gradient to be optimized only in one and the fastest direction, thereby enabling a universal and effective protection method.

The experimental comparison between utilizing vanilla and simplified optimization objective is presented in Section 6.4. Drawing from the aforementioned analyses, we select the pivotal objective $\mathcal{L}_{mel}$ as a core component of SafeSpeech.

Speech PErturbative Concealment. During the optimization process, we find that optimizing Eq. (5) can effectively make the synthesizer produce unclear speech. However, if we listen attentively, there are still parts of the slightly auditory pronunciation, and its background noise is relatively low. Moreover, we hope that the synthesizer can only generate noise when training on a SafeSpeech-protected dataset to achieve SA1 and SA2, and the robustness of the perturbation must be guaranteed. Based on this, we aim to propose a more effective and robust defense mechanism. When training the synthesizer, we expect the generator to produce noise. Therefore, in the perturbation optimization procedure, we utilize a Gaussian-distributed random noise z, leading the generator to produce noise. We aim that the distribution of the synthesized output can increasingly approximate a real noise distribution z. Based on this, we employ the Kullback-Leibler (KL) divergence as part of the optimization objective due to its *asymmetry*. Lower values of KL scatter mean that the output of the model is more similar to the noise distribution, which can achieve the low-quality of deepfake speech. At the same time, for both the random noise and generated audio, we extract mel-spectrogram feature z_{mel} and $\hat{x}_{mel}$, to acquire more relative information, and reduce the ℓ_1 distance as Eq. (5). The objective function can be described as:

$$\mathcal{L}_{noise} = D_{KL}(\hat{x}_{mel}, z_{mel}) + ||\hat{x}_{mel} - z_{mel}||_1, \tag{6}$$

where D_{KL} represents the KL divergence of two distributions.

Based on the above, we desire the model to learn the perturbations rather than speech information. To achieve this, we propose a method named Speech PErturbative Concealment (SPEC), which combines the mel function and the noise loss function assigning a suitable weight and can be expressed by:

$$\mathcal{L}_{SPEC} = \mathcal{L}_{mel} + \beta \mathcal{L}_{noise}. \tag{7}$$

where β is the hyperparameter to be set.

In conclusion, we introduce a pivotal objective function aimed at simplifying the multi-task learning problem to achieve effectiveness at the training stage and enhance transferability. Furthermore, we design the Speech PErturbative Concealment method that measures the two distributions between synthesized speech and random noise, thereby concealing the speaker's information with a stronger protection.

4.2 Perception Optimization

The perturbation should be generated without interfering with the normal exploitation of data samples. So the imperceptibility of the noise is an important factor. In the process of noise generation introduced in Section 4.1, ℓ_p norm is employed to limit the perturbation boundary so that the overall magnitude of the noise values can not be particularly large. However, the limitation of the perturbation in the value aspect cannot completely represent human perception. Based on this, SafeSpeech utilizes the noise perception module to reduce the gap between the ℓ_p norm and human perception. We optimize the audio perception in the time and frequency domains.

For better noise perception and imperceptibility, we employ Short-Time Objective Intelligibility (STOI) [70] score as our main part of the perception module. STOI score represents speech intelligibility as an objective metric, which computes the correlation of short-time temporal envelopes of the clean and protected audio, ranging from 0 to 1 and a higher score indicates better speech quality. STOI score is closely related to the human auditory perception and optimizing STOI function $\mathcal{L}_{stoi}$ brings a more natural sound. Moreover, we follow the principles to compute $\mathcal{L}_{stoi}$ introduced in [70].

On the other hand, we consider the time and frequency domain of audio for better perception optimization in the ℓ_p radius. Short-Time Fourier Transform (STFT) [72] performs well in feature extraction, so we utilize the ℓ_2 distance as part of our perception loss which can be expressed by:

$$\mathcal{L}_{stft} = ||\text{STFT}(x+\delta) - \text{STFT}(x)||_2. \tag{8}$$

Algorithm 1: SafeSpeech.

Inputs: $(x, text) \in D_c$, perturbation δ, surrogate model $\mathcal{M}$, optimization numbers max_epoch.
Parameters: random Gaussian noise z, ℓ_p norm boundary radius ε, weight coefficients α and β.
Output: protected audio x'.

$\delta \leftarrow$ `init_perturbation_set`$(-\varepsilon, \varepsilon)$;
$x' \leftarrow x + \delta$;
for $j \leftarrow 1$ **to** max_epoch **do**
  $\hat{x} \leftarrow \mathcal{M}(\texttt{spec}(x'), text, \texttt{other\_input})$;
  $\mathcal{C}_1 \leftarrow \mathcal{L}_{mel}(\hat{x}_{mel}, x)$;
  $\mathcal{C}_2 \leftarrow D_{KL}(\hat{x}_{mel}, z_{mel}) + \|\hat{x}_{mel} - z_{mel}\|_1$;
  if `Perception_Optimize` **then**
    $\mathcal{C}_3 \leftarrow \mathcal{L}_{perception}(x, x')$;
    $\mathcal{C} \leftarrow \mathcal{C}_1 + \beta \cdot \mathcal{C}_2 + \alpha \cdot \mathcal{C}_3$;
  else
    $\mathcal{C} \leftarrow \mathcal{C}_1 + \beta \cdot \mathcal{C}_2$;
  end
  $\delta \leftarrow$ `Clamp`$(-\texttt{sign}(\nabla_x \mathcal{C}), -\varepsilon, \varepsilon)$;
  $x' \leftarrow x + \delta$;
end
`evaluation_optimize_hyperparameters`().

Based on the above, the perception module of SafeSpeech crafts a hybrid optimization function:

$$\mathcal{L}_{perception} = \mathcal{L}_{stoi} + \mathcal{L}_{stft}. \tag{9}$$

Method Conclusion. Combining proposed data protection and perception optimization techniques, the objectives of SafeSpeech $\mathcal{L}$ can be expressed by:

$$\mathcal{L} = \mathcal{L}_{SPEC} + \alpha \mathcal{L}_{perception}, \tag{10}$$

where α is the weight coefficient the same as in Eq. (2), which balances the imperceptibility and effectiveness in Section 6.4.

Algorithm 1 shows the detailed description of SafeSpeech. The SafeSpeech can optimize the perturbation for max_epoch steps. `init_perturbation_set` assigns the perturbation to a random initial value within radius ε. If the effectiveness performance dissatisfies the user's expectation, the hyperparameter, such as max_epoch and ε, can be changed to enhance the protection performance until achieving a satisfactory level in the last step `evaluation_optimize_hyperparameters`.

To summarize, SafeSpeech achieves training-stage voice protection by introducing the pivotal objective and SPEC technique. In the pivotal objective, we innovatively select the function with the *fastest convergence rate and universality* to optimize, *i.e.* the $\mathcal{L}_{mel}$ in Eq. (5). In the SPEC technique, considering the *asymmetry* of the KL divergence, which can better measure the difference between real and synthetic distributions, we propose a speech concealing technique based on KL divergence to enhance the effectiveness, which is also novel compared to previous methods.

5 Experimental Settings

In this section, we describe our experimental settings on models, datasets, hyperparameters, and metrics. All the experiments were conducted on one NVIDIA A800 GPU.

5.1 Baselines

For a broader comparison, we consider two types of data protection: *perturbative availability poisons* (PAP) [39, 67], which protects data during the training stage, and *voice protection* techniques. Referring [39], we employ SOTA PAP baselines, including AdvPoison [15], SEP [11], and PTA [20]. In terms of voice protection, we utilize two open-source protection approaches: AntiFake [68] and AttackVC [19]. We provide a detailed comparison in Appendix A.

Adversarial Poisoning (AdvPoison) [15]. Fowl *et al.* [15] demonstrated that adversarial examples, particularly targeted attacks, can achieve more protective effectiveness.

Self-Ensemble Protection (SEP) [11]. The perturbation dynamically interferes with a DNN during its whole training process. Based on this, Chen *et al.* [11] proposed self-ensemble protection. It uses intermediate checkpoint models in a self-ensemble way to enhance and better simulate a training and dynamic model, improving perturbation generalization.

Patch-To-All (PTA). For VITS [29], MB-iSTFT-VITS [27] and BERT-VITS2, a fast and efficient training strategy, windowed generator training, is utilized to randomly crop a fixed-length sample from a complete audio. Drawing inspiration from comparable process methods employed in adversarial attacks, we devise an approach, Patch-To-All, which generates perturbation that minimizes the error from fragment audio [20] and patches it to the entire sample as a comparison.

AntiFake [68], **AttackVC** [19]. Given a clean sample x from speaker victim i, the speaker's timbre feature E_i is computed by an encoder. Subsequently, a targeted speaker j with the least similar timbre (in AntiFake) or randomly selected with the opposite gender (in AttackVC) is identified with timbre feature E_j. A perturbation is added to the original sample x in such a way that E_i becomes similar to E_j, thereby accomplishing voice cloning that results in timbre dissimilarity to the speaker victim i. We utilize the tools they have released to convert an original speech into a protected one.

5.2 Text-To-Speech Synthesizers

To comprehensively evaluate the effectiveness and transferability of SafeSpeech, we have selected a range of models for evaluation. On the one hand, we choose some classic, widely used, and improved models with fine-tuning capabilities in the TTS field. On the other hand, we select the latest and top-performing SOTA models based on the benchmark, TTSDS [3]. These models vary in architecture, encompassing those based on *generative flow* (GlowTTS [28]), *Variational*

Autoencoder (VAE) architectures (VITS [29] and MB-iSTFT-VITS [27]), *encoder-decoder* frameworks (OpenVoice [45]), *diffusion models* (StyleTTS 2 [36] and TorToise-TTS [7]), and *flow matching* (F5-TTS [12]).

Due to the remarkable capabilities of LLMs in dialogue and text generation, the advanced and latest TTS models, *e.g.*, BERT-VITS2 [1], XTTS [8], and FishSpeech [37], have generally integrated LLM components with synthesizers. In this setup, the LLM learns and emulates pronunciation characteristics and speaking styles of the target speakers, while the synthesizer focuses on learning and replicating the timbre features. This combination has led to a significant improvement in terms of synthetic naturalness. We provide a more detailed and comparative introduction to each model in Appendix B.

We utilize models with fine-tuning capabilities, *i.e.*, BERT-VITS2, StyleTTS 2, MB-iSTFT-VITS, VITS, and GlowTTS, to validate the protective effect at training time in Section 6.1, and zero-shot models, *i.e.*, TorToise-TTS, XTTS, OpenVoice, FishSpeech, and F5-TTS, to evaluate at inference time in Section 6.3. For GlowTTS, VITS, MB-iSTFT-VITS, and StyleTTS2, we use pre-trained models on LJSpeech [23] speech corpus. For BERT-VITS2, we utilize the model trained on a large-scale multilingual and multi-speaker speech corpus.

5.3 Experimental Datasets

We leverage utterances from two standard speech synthesis datasets for method assessment. LibriTTS [69] is utilized to evaluate the performance of our method in targeted single-speaker with long sentences, while CMU ARCTIC [30] is harnessed to assess multiple speakers with shorter sentences.

LibriTTS [69]. For effective fine-tuning, we select the top speaker who exhibits the highest similarity in voiceprint to the targeted speaker of LJSpeech. [23] from the LibriTTS train-clean-100 subset which is derived from LibriSpeech [44] corpus, a large-scale of speakers corpus.

CMU ARCTIC [30]. CMU ARCTIC includes audio recordings from 18 speakers and the speech content is nearly similar. For each speaker, we select 100 samples for fine-tuning. For the training dataset, we randomly shuffled each dataset and employed 80% of the audio samples for training, reserving the remaining 20% for evaluation. More detailed information on the two datasets is presented in Appendix B.

5.4 Hyperparameters and Metrics

In this part, we outline the hyperparameters in our experiments and evaluation metrics objectively and subjectively.

Hyperparameters. In the fine-tuning process, we keep the conventional hyperparameters in [27–29, 36] while setting the correct sampling rate in our customized dataset. In noise generation, we set the perturbation radius ε as $8/255$ to reach a balance of human perceptibility and unlearnability and optimize the noise until the perturbation performs well. To ensure the effectiveness of synthesis, we train the models 100 epochs for single speaker and 200 epochs for multi-speaker datasets. It is worth noting that we changed the input of the models from spectrograms to the original waveform to better fit the realistic training scenarios. We set the α in Eq. (10) as 0.05 to balance the imperceptibility and unlearnability and the β in Eq. (7) as set as 10 to achieve the best perturbative poisoning.

Metrics. We consider the subjective and objective metrics:

- *Mel-Cepstral Distortion* (MCD) [32]: MCD, using Dynamic Time Warping mode, measures the disparity in audio features between synthesized and real audio, reflecting differences in speech content, timbre, *etc*.
- *Word Error Rate* (WER) [25]: WER measures pronunciation clarity by a pre-trained medium-size Whisper [46] to recognize text. *Higher MCD and WER represent worse speech clarity to achieve synthesis quality protection*.
- *Speaker Similarity* (SIM) [25]: SIM is a metric to evaluate the timbre similarity between two speeches. Higher SIM represents the larger timbre similarity. We follow the principles from [25] and leverage ECAPA-TDNN [13] as the speaker encoder to compute the cosine similarity score between the real and synthetic speeches. When the SIM exceeds 0.25, personal voice has been successfully cloned in timbre [13]. The Attack Success Rate (ASR) is calculated by the ratio of successfully cloned samples in speaker similarity (*i.e.*, SIM > 0.25) to the total number of samples.
- *Signal-to-Noise Ratio* (SNR) [68]: SNR calculates the ratio between the embedded perturbation and the original audio to measure the levels of perturbation volume.
- *Naturalness* [6]: We utilize the advanced DNN-based audio predictor, UTMOSv2 [6] model, to evaluate the speech naturalness objectively. Moreover, we evaluate the naturalness subjective by human survey. *Higher values of SNR and Naturalness represent better speech quality.*
- *Mean Opinion Score* (MOS) [33]: MOS is a subjective evaluation metric that measures human perception of audio quality, typically ranging from 0 to 5, with higher values indicating better audio quality.

In conclusion, achieving higher MCD and WER values effectively fulfills SA2, preventing the malicious usage of synthesized samples. Simultaneously, maintaining a lower SIM adheres to SA1, realizing the identification protection.

6 Experiments and Analyses

For a comprehensive conclusion, we evaluate the effectiveness, transferability, and audibility of SafeSpeech. The effectiveness and transferability of SafeSpeech have been demonstrated against speech synthesis based on fine-tuning in Section 6.1 and zero-shot in Section 6.3, respectively. Additionally, we assess the naturalness and human perception of protected audio (in Section 6.1 and 6.2) and conduct a subjective evaluation to investigate the deceptive effects of synthesized

Table 1: Comparison of the TTS models trained on clean, random Gaussian noise added, patch-to-all (PTA), adversarial poisoning (AdvPoison), self-ensemble protection (SEP), AntiFake, AttackVC, and our proposed Speech PErturbative Concealment (SPEC) safeguarded dataset. The best and second-best unlearnability results are highlighted with **bold** and underlined, respectively.

Dataset	Method	BERT-VITS2 [1]			StyleTTS2 [36]			MB-iSTFT-VITS [27]			VITS [29]			GlowTTS [28]		
		MCD(↑)	WER(%)(↑)	SIM(↓)	MCD(↑)	WER(%)(↑)	SIM(↓)	MCD(↑)	WER(%)(↑)	SIM(↓)	MCD(↑)	WER(%)(↑)	SIM(↓)	MCD(↑)	WER(%)(↑)	SIM(↓)
D_1	ground truth	-	15.813	-	-	15.813	-	-	15.813	-	-	15.813	-	-	15.813	-
	clean	5.171	24.024	0.604	4.806	23.525	0.587	4.922	21.345	0.668	5.270	20.208	0.652	7.722	30.725	0.466
	random noise	5.444	27.747	0.472	5.457	23.617	0.466	5.397	24.654	0.489	5.503	33.267	0.449	9.853	49.675	0.311
	AdvPoison [15]	10.474	57.699	0.322	9.310	27.765	0.347	8.069	35.221	0.393	8.594	50.696	0.402	13.001	94.769	0.190
	SEP [11]	8.367	57.921	0.321	7.208	26.003	0.315	7.638	62.634	0.272	8.000	55.917	0.292	14.252	81.753	0.209
	PTA [20]	11.193	59.035	0.286	9.688	26.846	0.248	9.763	54.695	0.242	10.039	72.304	0.240	17.889	85.018	0.144
	AttackVC [19]	6.376	31.141	0.525	4.660	20.419	0.419	5.665	27.384	0.527	5.188	29.940	0.638	8.939	62.572	0.289
	AntiFake [68]	7.740	48.966	0.254	7.755	42.890	0.214	6.748	58.420	0.234	6.164	63.604	0.221	12.341	98.410	0.090
	SPEC (ours)	**14.771**	**99.610**	**0.204**	**10.278**	**57.693**	**-0.011**	**13.826**	**94.706**	**0.172**	**11.566**	**93.270**	**0.178**	**22.093**	**102.407**	**0.081**
D_2	ground truth	-	8.290	-	-	8.290	-	-	8.290	-	-	8.290	-	-	8.290	-
	clean	5.629	21.658	0.658	5.079	6.660	0.561	5.709	9.700	0.588	5.591	12.460	0.626	7.702	30.186	0.425
	random noise	6.012	26.330	0.570	6.275	10.497	0.466	6.103	11.794	0.469	6.168	16.491	0.516	9.586	37.921	0.343
	AdvPoison [15]	10.438	37.924	0.398	10.257	13.775	0.292	9.150	28.340	0.359	9.286	52.709	0.349	14.207	87.318	0.072
	SEP [11]	8.284	50.569	0.433	8.405	14.347	0.289	8.390	32.622	0.322	8.768	46.412	0.338	14.423	76.617	0.118
	PTA [20]	11.504	46.619	0.365	9.470	15.961	0.368	11.041	29.436	0.249	12.040	57.050	0.280	17.835	82.882	0.084
	SPEC (ours)	**15.175**	**80.291**	**0.273**	**12.303**	**16.967**	**0.267**	**13.631**	**54.763**	**0.206**	**13.387**	**72.909**	**0.243**	**19.646**	**96.279**	**0.069**

D_1 and D_2 represent LibriTTS and CMU ARCTIC datasets, respectively.

speech on humans. Finally, we perform ablation studies focusing on the method components and the hyperparameters.

6.1 Effectiveness and Transferability

The assessment of SafeSpeech's effectiveness encompasses three different stages: the perturbation generation on the surrogate model, training on safeguarded audio samples by different methods, and evaluation of the synthetic performance.

Perturbation Generation. We select BERT-VITS2 as the surrogate model to generate perturbation due to its superior performance in high-quality speech synthesis and fine-tuning capability. We utilize the surrogate model to generate perturbation via SafeSpeech and evaluate its transferability on other SOTA TTS models. For a comprehensive evaluation, we select six baseline methods, including random noise and the specifically generated perturbation.

Training on the Safeguarded Dataset. After acquiring specific noises from the surrogate model, we apply the perturbation to the original audio, creating the protected dataset. Fine-tuning on unprotected audio samples results in a plausible speaker synthesizer. We compare this scenario with SafeSpeech-protected audio and verify their unlearnability across StyleTTS2, MB-iSTFT-VITS, VITS, and GlowTTS without *a priori* knowledge of the model structures.

Speech Synthesis and Evaluation Results After training, we assess the synthesizer's performance on the test set. By inputting a speaker ID and text, the synthesizer produces realistic deepfake audio. For each test sample, we supply the generator with speaker ID and text, yielding synthesized speech. We evaluate it by measuring MCD and SIM between real and synthesized speech and using WER to measure speech clarity.

Effectiveness and Transferability Analyses. Table 1 shows the experimental results. Our proposed SPEC has achieved excellent protection performance on both single-speaker and multi-speaker datasets in terms of timbre (SIM) and speech intelligibility (MCD and WER). On the LibriTTS dataset, the SPEC effectively safeguards speeches from being learned with a significant increase in WER from 24.024% of the clean dataset to 99.610%. High WER values represent low speech quality (SA2). And SIM is the lowest at 0.204 which satisfies SA1. Moreover, the results show a broad range of transferability across TTS models with distinct structures. Compared to the PAP methods for data protection during the training stage and voice protection techniques for the inference stage and speaker similarity, the SPEC approach demonstrates superior performance in preventing the usage and high similarity of the synthesized audio with outstanding transferability. The reasons encompass two aspects. (1) The *pivotal objective optimization* is a universal objective, ensuring better effectiveness and transferability. (2) The *SPEC* technique can effectively conceal the speaker's information, thereby successfully preventing the model from learning audio samples.

Table 2: Objective evaluation of the similarity and naturalness between protected and original real audio samples.

	AdvPoison	SEP	PTA	AntiFake	AttackVC	**SafeSpeech**
Similarity(↑)	0.715	0.703	0.776	0.719	**0.974**	0.859
Naturalness(↑)	3.343	2.571	2.515	2.824	**4.289**	3.021

Perception Analyses. In the design of SafeSpeech, we optimize the perception of the perturbation in the time and frequency domain (Section 4.2). It is crucial that the perturbation cannot affect the normal use of the audio or alter the timbre. Therefore, we objectively evaluate the best speaker similarity (the SIM metric) and naturalness between the protected and clean audio. Table 2 illustrates the results of our experiments, which show that compared to better-performing baselines, *e.g*, PTA and AntiFake, SafeSpeech achieves a similarity score of 0.859, indicating almost no alteration in the timbre, and a naturalness score of 3.021. Moreover, we balance the effectiveness and perception by sampling α in Section 6.4.

Table 3: The subjective evaluation of the ground truth (GT) and synthesized speech.

	MOS(↓)
GT	4.756 ± 0.103
clean	4.677 ± 0.114
PTA	2.008 ± 0.191
SPEC (ours)	$\mathbf{1.070 \pm 0.161}$

Table 4: Human perceptual evaluation of the similarity and naturalness between protected and original real audio samples.

Similarity	**Naturalness**
98.333%	3.190 ± 0.189

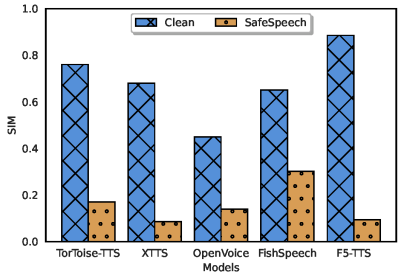


Figure 4: Speaker similarity of synthesized samples after zero-shot voice cloning on the clean and protected audio.

This experiment confirms that SafeSpeech is effective and transferable with minimal alteration of the original audio.

6.2 User Study

In the real world, deepfake speech usually needs to deceive human victims. Therefore, in this section, we explore the human perception of protected samples and synthetic speech.

Preliminary Work. The Human Ethics Research Committee affiliated with the primary author determined that this study was exempt from further human subject review. We created the anonymous questionnaire and recruited participants through the Credamo platform.

Participants. We recruited 80 participants (after filtering in Appendix C), all of whom were between 18 and 40 years old and had proficient English skills. Before participating, we obtained their consent and provided an average compensation of $0.30 per participant. Their average spent time is 241.075 seconds, providing reliable subjective results.

Study Setting. To throughout evaluate, we have designed three parts in each questionnaire with 23 questions to explore the synthesis quality, speaker similarity between protected and original audio, and the naturalness of protected audio.

Part 1: Synthesis Quality. We selected three ground truth and synthesized audio from the clean, better-performing baseline (PTA), and SafeSpeech-protected datasets (twelve samples in total) to validate the subjective quality. The participants rated the audio quality on a scale from 0 to 5. Table 3 shows the results. Clean synthesized samples have the MOS value of 4.677 ± 0.114, indicating good audio perceptual quality and the potential to deceive participants. In contrast, the MOS for the audio synthesized from the SafeSpeech-protected dataset is much lower at 1.070 ± 0.161, reflecting the poor audio quality and inability to deceive participants effectively.

Part 2: Speaker Similarity. The experiments in Part 1 demonstrate the perceptual effectiveness of SafeSpeech and we also consider if the protected audio retains a similar timbre to the original. We select three pairs of protected audio and original audio with one pair from different speakers for the experiment. Table 4 shows that 98.333% of participants believe the protected audio came from the same speaker as the original, indicating minimal alteration to the original speaker's timbre.

Part 3: Naturalness of Protected Audio. We aim to degrade the detectability of human perception. In this part, we ask participants to rate the naturalness of the protected audio from 0 to 5 [68]. Table 4 shows that the naturalness score for the protected audio is 3.190 ± 0.189. Generally, a score above 3 is considered to indicate relatively high quality [68]. Therefore, most participants find the protected audio is natural or the embedded perturbation is acceptable.

6.3 Defense against Zero-Shot Voice Cloning

When adversaries derive the audio samples of the target speaker, they may apply them to fine-tune a synthesizer or conduct zero-shot cloning with a limited number of samples. Zero-shot voice cloning requires fewer computational resources than fine-tuning but degrades the synthetic result. The experiments in Section 6.1 demonstrate SafeSpeech's effectiveness at training time. However, we also aim SafeSpeech can still perform well against zero-shot voice cloning. In this section, we utilize five advanced SOTA TTS models, *i.e.*, TorToise-TTS [7], XTTS [8], OpenVoice [45], FishSpeech [37], and F5-TTS [12], with outstanding zero-shot capability to evaluate the synthesis on two clean and protected samples, respectively.

Experiments and Results. Figure 4 presents the speaker similarity, *i.e.*, the SIM metric, between the clean and SafeSpeech-protected synthesized speech to real audio. Surprisingly, although the generation process of perturbations does not depend on these zero-shot models, and the perturbation is specifically considered for the training phase, SafeSpeech can still protect our speech during the inference procedure on the unseen and advanced models. On the F5-TTS, the SIM value drops from 0.885 for clean samples to 0.094 for protected samples. Even the FishSpeech, which exhibits the best performance against noise, achieves only a SIM of 0.301 on protected samples. This indicates that SafeSpeech remains effective in defending against zero-shot voice cloning.

Analyses. This effectiveness is due to our proposed SPEC method in Section 4.1, which leverages the surrogate model to guide the synthesized speech more noise-like via the KL divergence, thereby concealing the original speaker's information. Consequently, this approach ensures protection during both the fine-tuning and zero-shot stages.

Table 5: The difference between the vanilla and our proposed pivotal function as optimization function.

Method	# Params	Vanilla		**Pivotal (ours)**	
		MCD(↑)	WER(↑)	MCD(↑)	WER(↑)
BERT-VITS2	104.64 M	10.316	72.435	**10.722**	**81.517**
MB-iSTFT-VITS	80.78 M	9.074	64.421	**9.945**	**73.756**
VITS	82.42 M	9.773	64.614	**10.419**	**87.846**
GlowTTS	32.03 M	17.113	95.632	**18.607**	**104.887**

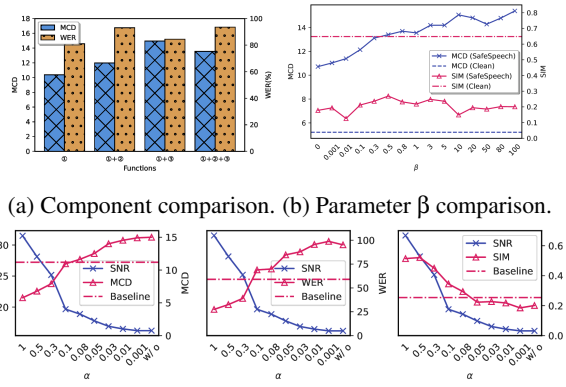
6.4 Ablation Study

In Section 4.1, we delve into the rationale behind objective selection. Building upon this foundation, this section presents a comprehensive comparison of the original and pivotal objective optimization methods, evaluating their effectiveness and runtime performance in perturbation optimization. We achieve a balance of effectiveness and perception of perturbations by sampling α in Eq. (10). Meanwhile, our proposed SPEC is a multi-task learning problem, so the setting of β is an issue of interest. We conduct the ablation study on the LibriTTS dataset and the BERT-VITS2 model.

Efficiency and Effectiveness of Pivotal Objective. In Section 4.1, we have introduced the problem formulation and illustrated the comparison between vanilla and pivotal unlearnable examples. To address a multi-task optimization problem, we simplify it into a single-task optimization problem and illustrate the principles of the pivotal function selection strategy. Taking the BERT-VITS2 model as an example, through the analysis of the convergence speed, we devise a universal function that is most relevant to the audio content as part of our target optimization function. This section compares using the $\mathcal{L}_{mel}$ as the primary objective of optimizing the entire generator functions in terms of efficiency and time cost.

Table 5 shows our experimental results. Among them, on BERT-VITS2 we find that the WER increases from 72.435% to 81.656%, which means less clear speech expression and noisy background. We find that all have certain unlearnability across three models. While achieving better protection, choosing the pivotal function to optimize can greatly shorten the noise optimization runtime. In our experiments, the time for optimizing one identical sample to generate vanilla perturbation is 10.3 seconds, whereas the time required for pivotal optimized perturbation is 4.0 seconds, resulting in a nearly 61.2% reduction which can be employed for real-world application. The improved efficiency is due to the pivotal function optimization can avoid useless calculations.

Components Analyses. The objective function of SPEC is described as Eq. (7) which in detail can be divided into three parts: ① the pivotal function; ② the KL divergence using Gaussian noise to lead the noise-like output; ③ the ℓ_1 norm between random noise and synthetic audio. To investigate how each function affects the protective effect, we carry out the ablation study on the LibriTTS dataset, considering the combination of functions: only ①, ①+②, ①+③ and ①+②+③,



(a) Component comparison. (b) Parameter β comparison.

(c) Comparison of perception and effect across different α.

Figure 5: Ablation study about components analyses and hyperparameter settings on different evaluation metrics.

respectively. The results in Figure 5a demonstrate the different effectiveness of the four functions. Among them, the introduction of noise-leading methods in ② and ③ both yield better results than using the pivotal loss function ① alone. Additionally, we find that the combination of the three functions, ①+②+③, performs best on WER and the SIM value is only 0.21, which is lower than the speaker similarity threshold, indicating outstanding protection of the speaker's timbre when combining the three functions as Eq. (7).

Balance of Strength and Perception. In Section 4.2, we introduce a perceptual loss based on two evaluation metrics for perturbation, *i.e.*, STFT and STOI, to enhance the imperceptibility in the time and frequency domain. Eq. (10) shows the optimization objective of SafeSpeech, while the value of α influences the effectiveness of protection and imperceptibility of the perturbation: a larger α results in better auditory quality but weaker protective effect. We explore the balance by sampling α from 0.001 to 1 with the SNR as the perceptual metric. Figure 5c illustrates the results. We can find that when α is set to 0.05, the effectiveness metrics are higher than "Baseline" (here we select PTA), *i.e.*, resulting in an MCD of 12.516, a WER of 84.709%, and a SIM of 0.223. The perceptual metric increases from an initial 16.021 to 17.791, which also surpasses the "Baseline" score of 16.578.

Hyperparameter Study. Our proposed objective Eq. (7) shows a multi-task optimization problem, in which the value of the weight coefficient β has a certain impact on the performance of noise optimization. In this experiment, we study how different values of β impact the unlearnability of protected audio. We carry out experiments on a single speaker from LibriTTS and fine-tune the utterances for 100 iterations. We establish a range for β from 0 to 100, with various intervals, resulting in a total of fifteen distinct β values. Figure 5b shows the results of different β on the MCD and SIM metrics. We observe that the impact of unlearnability remains consistent across different values of β, enabling us to achieve

satisfactory results regardless of its specific setting. We find that when β is set to 0.01 or 10, the timbre is protected achieving well, while the value of 10 also poisoning the dataset with a higher MCD of 15.060 compared to 11.356. On this basis, we have chosen β to be 10, which yields WER and SIM values of 95.266% and 0.149, respectively. When compared to the training on clean samples, the MCD and SIM values stand at 5.217 and 0.648, respectively, thus demonstrating a remarkable protection performance. The insensitivity of the protective effect to the hyperparameter indicates the stability.

7 Robustness against Adaptive Attackers

SafeSpeech possesses high robustness against strong adaptive adversaries. In this section, we consider and conclude three levels of adversaries: (1) *Data Level*. Data-level technologies include perturbation removal in Section 7.1.1, advanced data augmentation in Section 7.1.2, and optimization-based speech recovery in Section 7.1.3. (2) *Model Level*. Adversaries may employ model recovery in Section 7.2.1, robust training in Section 7.2.2, and fine-tuning with clean data in Section 7.2.3. (3) *Real-world Level*. Protection in the physical world is also a requirement at a higher level. We will evaluate the robustness of SafeSpeech in the physical world in Section 7.3, as well as the performance and time overhead in real-time scenarios.

7.1 Data-Level Robustness

7.1.1 Advanced Perturbation Removal

The embedded perturbation for protecting audio by SafeSpeech may be detected by adversaries, and they can remove this noise to improve the synthetic performance. In this section, we consider the traditional denoising technique, spectral gating (SG), as well as the current mainstream and advanced denoising model based on deep learning, DEMUCS [49].

Traditional Denoise. SG aims to remove the relatively low value in the time domain of the speech. After training with SG denoising on audio protected by SafeSpeech, the results show a WER of **69.321%**, and a SIM of **0.233**, indicating that SafeSpeech remains relatively effective under this condition.

Advanced Denoise. DEMUCS can effectively eliminate perturbations and obtain approximately clean samples without a noisy background. We train the model using DEMUCS-denoised samples, resulting in a WER of **57.329%** and a SIM of **0.284**. This indicates that the synthesized speech is dissimilar and of low quality compared to the original samples.

Reason and Analyses. SafeSpeech is robust against traditional and advanced perturbation removal techniques. This is because SafeSpeech embeds imperceptible perturbations, while denoising can remove noise along with some original speaker information, *e.g.*, timbre, and phoneme features.

Table 6: The robustness quantization via data augmentation and defensive methods. The underline values indicate the most significant decreases in protection compared to training without data augmentation, *i.e.*, SPEC ("w/ o" in the Table).

Metric	w/ o	Defense-based [21]				Transformation-based				Diffusion-based
		RS	Mel	QD	FL	Speed	Mask	LPF	MP3	AudioPure [64]
MCD(↑)	14.771	14.368	14.679	14.486	13.222	11.455	14.983	13.991	15.097	14.216
WER(↑)	99.610	96.781	99.149	91.363	97.306	97.444	100.899	98.082	93.545	85.711
SIM(↓)	0.204	0.168	0.179	0.238	0.227	0.106	0.247	0.252	0.261	0.227

7.1.2 Data Augmentation

In the real world, the attackers may adopt various data augmentation methods to destroy the specific perturbation to improve the model performance, so we hope that users' uploaded audio protected by SafeSpeech can retain the consistency of unlearnability when facing real-world speech synthesis attacks with different data augmentation methods. For a comprehensive evaluation, following [10, 68], we consider three categories of effective data augmentation techniques: defense-based, transformation-based, and diffusion-based.

- **Defense-based Techniques**: These include down-sampling and up-sampling (RS), mel-spectrogram extraction and inversion (Mel), quantization-dequantization (QD), and frequency filtering (FL), which are designed to effectively prevent adversarial audio examples from WaveGuard [21].
- **Transformation-based Techniques**: This category encompasses speed adjustment (Speed), time masking (Mask), low-pass filtering (LPF), and MP3 compression, which are commonly applied in real-world audio processing.
- **Diffusion-based Techniques**: AudioPure [64]. AudioPure aims to disrupt the perturbed audio by diffusion model, which represents the SOTA audio defense technique.

From the results in Table 6, we can observe that these data augmentation techniques diminish the protective effect to a certain extent. For instance, the WER decreases to 85.711% by AudioPure, and the SIM metric drops to 0.261 by MP3 Compression. However, compared to clean samples or those with random noise, the protection remains effective. This indicates that SafeSpeech demonstrates robustness against audio data augmentation techniques. The reason for this robustness is that while data augmentation can disrupt the structures of embedded perturbations, these transformations can also degrade speech quality (*e.g.*, MP3 Compression, mel extraction, and speed adjustment). Consequently, lower-quality speech samples as input will result in the degradation of both the quality and similarity of the synthesized speeches.

7.1.3 Optimization-based Speech Recovery

The adversaries may attempt to recover the original speech from the perturbed state in a "reverse" direction as SafeSpeech, with the critical challenge of determining the "reverse" optimization direction. However, the optimization direction

remains unknown due to the lack of clean audio from the original speaker. Further analysis reveals that SafeSpeech aims to conceal the speaker's privacy by embedding perturbations. Therefore, the adversaries may leverage two types of feedback to determine the optimization direction [10, 68]: (1) The naturalness score, aiming to restore it to an unperturbed state; (2) By increasing adversaries' capabilities, they can query a speaker recognition system enrolled with the target speaker, to recover the hidden characteristics.

Following [9, 22], we employ a black-box optimization method, Natural Evolution Strategies (NES), with a total of 50000 queries to optimize a random sample from the LibriTTS dataset. Then, the reversed sample is cloned by FishSpeech, resulting in a SIM value of 0.252. Compared to the initial perturbed sample of a SIM of 0.168, this optimization-based method can improve the synthetic performance but is still far from the original sample's SIM value of 0.561. The reasons lie in two aspects. Firstly, adding perturbation during this optimization degrades the speech quality. Secondly, the "reverse" direction is estimated and not inaccurate.

7.2 Model-Level Robustness

7.2.1 Model Recovery

In this experiment, we verify the necessity of fine-tuning for the target speaker and the feasibility of the adversary's model recovery technique by effectiveness comparison.

Before performing speech synthesis, the adversary already possesses a model pre-trained on a large-scale dataset, and subsequently acquires a protected dataset for fine-tuning, with the hope of cloning the target speaker. However, after training, they are unable to clone high-quality audio and realize that the model may have been perturbed. Consequently, they can *recover from using the original model* for voice generation, so this experiment considers the model scenario and parameter restoration. We fine-tune the pre-trained model, BERT-VITS2, and validate the untrained synthetic voice against the LibriTTS speaker. The results are an **MCD of 14.911, a WER of 100%, and a SIM of 0.049**, indicating that without fine-tuning, the generated audio is not similar compared to fine-tuning of MCD at 5.171, WER at 24.024%, and SIM at 0.604. Therefore, direct synthesis without fine-tuning cannot achieve an effective synthetic result for the target victim.

We also find that the WER value is relatively high, which is because the released BERT-VITS2 model is trained on a large-scale dataset without setting the exact speaker number, while our focus is on fine-tuning for a single speaker. Therefore, it cannot completely load the weight files; only by fine-tuning the pre-trained model can we generate audible audio.

7.2.2 Advanced Robust Training

Previous PAP methods show the vulnerability against robust training techniques [16, 20]. In this section, we employ adversarial training to illustrate the robustness of SafeSpeech.

Table 7: The model performance across different defensive perturbation radius ρ_u and adversarial perturbation radius ρ_a.

ρ_a	$\rho_u = 8/255$			$\rho_u = 4/255$		
	MCD(↑)	WER(%)(↑)	SIM(↓)	MCD(↑)	WER(%)(↑)	SIM(↓)
0	14.771	99.610	0.204	10.921	76.110	0.302
2/255	14.592	92.891	0.214	8.538	55.581	0.292
4/255	12.361	99.079	0.147	**6.554**	**55.127**	**0.315**
8/255	11.323	**82.504**	0.188	7.946	68.467	0.246
10/255	**11.060**	94.608	0.198	8.621	80.703	0.225
12/255	11.771	107.829	**0.220**	9.086	84.630	0.243
16/255	12.262	113.238	0.193	10.304	94.940	0.237

Adversarial Training. Adversarial training aims to generate adversarial perturbations by maximizing errors and incorporating them into training samples to enhance the model's robustness and performance. After the adversary obtains the protected samples, they can maximize the objective function in Eq. (10) to generate adversarial perturbation that can mitigate SafeSpeech. Meanwhile, previous perturbative defense mechanisms that generate perturbations based on ℓ_p norm are vulnerable to attacks if the adversarial perturbation radius ρ_a exceeds the defensive data perturbation radius ρ_u. In such cases, the effectiveness of data protection significantly degrades, making non-robust data protection methods easy to compromise. In this section, for a comprehensive evaluation of the robustness of data protection, we set ρ_u to $8/255$ and $4/255$ respectively, while the ρ_a is varied across a range of values: $0, 2/255, 4/255, 8/255, 10/255, 12/255, 16/255$, to conduct a thorough assessment of the adversarial training.

From Table 7, we can find that when ρ_a is greater than or equal to ρ_u, the data protection effect can be reduced. For example, when ρ_a and ρ_u are both $8/255$, MCD and WER decreased from 14.716 and 97.090% to 11.323 and 82.504% respectively. However, the attack result of model performance improvement is still not obvious, far higher than the threshold. This proves that in the face of adversarial training, whether it is less than, equal to, or greater than the ℓ_p radius, the attacker still cannot maliciously clone the protected data.

7.2.3 Fine-tuning with Clean Data

In this section, we explore if SafeSpeech can be circumvented by clean-data fine-tuning and analyze its difference from data poisoning. Adversaries on the Internet may obtain both clean and SafeSpeech-protected audio samples and employ the mixed data for training to boost efficiency. If the model performs poorly on a certain speaker after training, it may suggest the speaker is protected. Then, they may use clean-data fine-tuning to reduce the speaker's impact on the model and try to recover the model performance.

Training with mixed data. We randomly select five speakers from the LibriTTS dataset [69], where speaker i is protected, and the remaining four are clean speakers, forming a mixed dataset for training on BERT-VITS2. The protected samples

account for approximately 14.6% (108 out of 738) of this dataset. After training, the results present that cloning speaker *i* yielded a WER of 108.336% and a SIM of 0.221 while testing with clean speakers resulted in a WER of 16.506% and a SIM of 0.686. This indicates that during joint mixed training, the protected samples do not interfere with the clean samples, nor do the clean samples mitigate the protected samples. Therefore, SafeSpeech only affects the audio that needs protection, which is distinctly different from data poisoning. Data poisoning aims to degrade model performance, meaning perturbed audio would affect the clean samples.

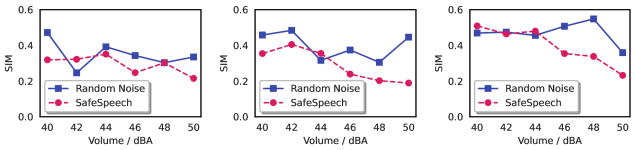
Fine-tuning with clean data. The aforementioned scenario illustrates that audio protected by SafeSpeech is not mitigated by clean samples trained alongside it. Therefore, adversaries may obtain additional clean samples for further fine-tuning to mitigate the perturbed samples and recover the model's performance. We randomly select five new speakers from the LibriTTS dataset to form a clean sample dataset for fine-tuning, with a total of 616 training samples. After fine-tuning, we test models on speaker *i* and result in a WER of 37.050% and a SIM of 0.126. It can be observed that fine-tuning with clean data leads to some improvement in clarity, but the similarity is highly low. This is because fine-tuning with clean data is conducted after training speaker *i* which means fine-tuning overwrites the original speaker's timbre and replaces it with the characteristics of the fine-tuned samples, *i.e.*, previous speakers have not been "learned" to achieve effective cloning.

7.3 Real-World Robustness

In real-world scenarios, *e.g.*, personal on-site presentations or online live broadcasts, (near) real-time and effective protection is required. In this section, we test the robustness and time overhead under real-time requirements in the real world. We utilize a lightweight TTS model, MB-iSTFT-VITS [27], as the surrogate model for perturbation generation.

Universality. For a single speaker, we generate perturbation from an audio sample and apply it to pad or truncate other samples. After fine-tuning with MB-iSTFT-VITS, the WER is 72.472%, and SIM is 0.197, indicating SafeSpeech can protect using just one segment of the target speaker's audio.

Real-World Protection. We invite a volunteer to read LibriTTS texts in a quiet room for 10 minutes for each test. The initial volume of the room is 22 dBA. We deploy SafeSpeech on a GPU-equipped device as the back end. The front end is a Lenovo laptop with an Intel(R) microphone for recording and a Lenovo BMS09 speaker for playing noise as the defender. The whole process is: When the microphone captures about 5 seconds of audio containing the target speaker, the front end sends the recorded audio to SafeSpeech's back end (*speaker → microphone → SafeSpeech*). SafeSpeech then generates perturbations from it and continuously sends perturbations to the front-end speaker for playback (*SafeSpeech → speaker*), thus achieving real-time protection in the real world.



(a) FishSpeech. (b) XTTS. (c) F5-TTS.

Figure 6: The results of synthetic speaker similarity in the physical world across three TTS models and volumes.

Given that recording in real-world scenarios may suffer quality degradation, we apply SG denoise and "loud-norm" [59] to the audio received by SafeSpeech to enhance vocal clarity. Meanwhile, an adversary records the live sound from a distance of 50 cm using a mobile device, Android VIVO.

Results and Analyses. Referring to VSMask [61], we evaluate the protection effectiveness of perturbations at volumes ranging from 40 dBA to 50 dBA, with random noise added as a reference. We test the volumes for ten seconds and calculate the averages. We conduct voice cloning tests on three models, *i.e.*, FishSpeech [37], XTTS [8], and F5-TTS [12]. Figure 6 illustrates the performance in the real world. It shows that at 40 dBA, SafeSpeech can resist cloning attempts of FishSpeech and XTTS, whereas, at 50 dBA, these three models cannot effectively clone the speaker with a SIM of 0.215 in FishSpeech. Moreover, we find that SafeSpeech outperforms random noise. This experiment demonstrates SafeSpeech's robustness in the real world. The reason for robustness lies that real-world recording is equivalent to a transformation, and SafeSpeech can resist data augmentation (Section 7.1.2).

Time Overhead. We build SafeSpeech on an NVIDIA A800 GPU device and average results over ten runs for reliability. The whole process, from getting the initial audio to complete playback, takes 13.898 seconds. It takes 10.606 seconds to generate perturbation for the speaker and 0.369 seconds to transmit the noise via various devices and networks. Compared to VSMask [61], a real-time voice defender that takes about 300 seconds to predict a speaker, SafeSpeech can protect on average in just a 14-second lead time for continuous protection. *The outstanding real-time capability comes from the pivotal function of decreasing the computational time and the choice of a lightweight surrogate model.*

8 Discussions and Limitations

In this section, we discuss some unavoidable points.

Distinction from Data Poisoning. We aim that the protected data cannot be learned by the TTS models. The experiment in Section 7.2.3 illustrates this point. If an adversary unauthorizedly obtains the target speaker's voice in a batch of data, this batch of data cannot be learned or cloned, and it does not affect the use of other authorized data. Data poisoning aims to degrade the model's performance. Although it protects unau-

thorized use to some extent, it also interferes with the use of authorized data, thus affecting the right of other data to be used, which is not our intention. Although SafeSpeech is like data poisoning, our purpose and results are quite different.

Benifits. Compared to adversarial-example-based voice protection techniques [19, 61, 68], we propose the pivotal objective optimization based on unlearnable examples that can effectively achieve training stage protection with a broader application rather than zero-shot. Compared to PAP methods [11, 15, 20], we introduce the SPEC technique based on KL divergence to guide the model output towards noise audio with the actual speaker information seemly "Consealing".

Further Effectiveness Enhancement. In the future, we can improve the effectiveness of SafeSpeech by two measures. First, since the generation of perturbations is constrained by ℓ_p norm, increasing the perturbation radius can yield better effects, as proven by the experiment in Appendix D.3. Simultaneously, we can improve the acceptability of the perturbation by proposed STFT and SIOT metrics. Secondly, utilizing the surrogate model can be considered. From the experiment in Appendix D.2, we can find that the specific perturbations generated on this model slightly outperform transferability-based samples. Therefore, to improve the effectiveness of unknown models, an ensemble of models can be employed as [68], although this will come with a significant computational cost.

Broader Protective Strength. (1) *Effectiveness*. We have evaluated the effectiveness of SafeSpeech under fine-tuning and zero-shot scenarios on single and multiple-speaker datasets. (2) *Transferability*. We use one surrogate model to protect the dataset and validate the transferability across the other ten models. (3) *Robustness and Real-Time*. We have considered a wide range of robustness in data, model, and real-world levels and confirm the real-time capability under speaker $\rightarrow$ microphone $\rightarrow$ SafeSpeech $\rightarrow$ speaker chain.

Long and Complex Audio. In daily utilization, users may aim to protect much longer audio samples. SafeSpeech has scalability and can handle longer and more complex audio. In Section 7.3, the volunteer read audio that lasted about 10 minutes each test. Due to the universality of SafeSpeech, the generated perturbations can be scaled to longer audio. We have also demonstrated this point with the audio volume of approximately two hours for the two datasets in Section 6.1. Section 7.3 also demonstrates that the transmission time of the perturbations on different devices is only 0.369 seconds each time, which is acceptable. For more complex audio, *e.g.*, containing significant noise, TTS models tend to produce lower-quality outputs even without protection by SafeSpeech.

9 Related Work

9.1 Audio Privacy Preservation

Current voice privacy-enhancing techniques also include speaker anonymization [14, 40] and audio watermarking [71]. Speaker anonymization aims to protect the speaker's identity in voice data while preserving the speech content [53]. Physical anonymization [17] can be used to isolate the original speech, while logical anonymization [66] bypass the authentication system. Audio watermarking aims to protect the audio copyright [42], content authentication [5], timbre certification [38], *etc.*, without altering the original audio quality by embedding specific information. However, these methods cannot achieve SA2 with high-quality synthesis.

9.2 Perturbative Availability Poisons

PAP techniques are designed to prevent models from learning (*e.g.*, by adding perturbations to the data) [20, 39, 60]. Huang *et al.* [20] found that the model learns the embedded error-minimizing noise rather than the information on clean labels. Building on this, Fowl *et al.* [15] utilized adversarial examples for more effective data poisoning. Yu *et al.* [67] generated linearly separable Gaussian perturbations in the ℓ_2 plane. These approaches have achieved SOTA PAP in the classification tasks [39]. Models may incorrectly assume that the data is not worth learning during the learning process due to the effect of specific clean-label noise, and thus discard noisy data due to fitting problems, resulting in unlearnable datasets. However, unlearnable examples [20] are fragile to data augmentation [63] and robust training [16], which can weaken the poisoning effect on unlearnable examples.

10 Conclusion

In this paper, we propose a proactive defense framework, SafeSpeech, to protect our voices from unauthorized speech synthesis via embedding imperceptible perturbations on original speeches before publicly releasing them. Extensive experimental evaluation shows that SafeSpeech has the most advanced voice protection effect to date, sufficient transferability to face various TTS models with distinct structures and backbones, and can resist the strength of various adaptive attackers in the real world. Moreover, SafeSpeech can effectively achieve real-time voice protection under scenarios of personal on-site presentations and reduce the security threats brought by voice cloning in the real world.

Acknowledgements

We sincerely thank the anonymous reviewers and the shepherd for their constructive feedback on our work. This research is supported in part by the National Natural Science Foundation of China under Grant No. U21B2020, the Beijing Natural Science Foundation under Grant No. QY24213, the Fundamental Research Funds for the Central Universities under Grant No. 2024ZCJH05, and the National Natural Science Foundation of China under Grant No. 62202064.

Ethics Considerations

We pay great attention to the potential safety issues that various research in society may raise, including those arising during the experimental phase and the release of SafeSpeech for use. In this paper, we strive to mitigate ethical concerns.

Subjective Consideration. A crucial purpose of our method is to prevent individuals from being deceived by deepfake audio, making human interaction experiments particularly important. Before conducting experiments, we considered ethical implications and sought opinions from relevant entities. The primary author's affiliated Human Ethics Research Committee determined that this study was exempt from further human subject review. All participants are over 18 years old, and we have sought their consent before experimenting. Throughout the experiment, no additional information was collected from the participants; all responses were anonymized. At the same time, we have informed them in advance that the content in these audios (LibriTTS dataset [69]) comes from an audiobook, and it is not a real event. Finally, the fake audio produced by this experiment (especially high-quality usable audio) is only used for the research of this experiment, not for other research, and these deepfake speeches were abandoned after the research was carried out.

Legitimate and Beneficial Usage. The initial intention of designing SafeSpeech is to prevent malicious speech synthesis and protect personal voice privacy. However, we recognize that speech synthesis can also be legitimate and beneficial, such as for disabled individuals who require speech synthesis tools. Therefore, SafeSpeech should not impede positive speech synthesis. Experimental results detailed in Section 7.2.3 demonstrate that training with a mix of protected and clean audio does not affect the synthesis quality of unprotected voices by SafeSpeech. Moreover, we will release SafeSpeech by authorized request. If users want to utilize SafeSpeech to protect their voices, they need to obtain our written usage authentication and fill in the usage rules of SafeSpeech, which state that legitimate and beneficial uses of TTS tools are not allowed to be perturbed. Meanwhile, they should sign relevant **disclaimer clauses** to ensure that their usage behaviors are not related to the designer and publisher of SafeSpeech.

Open Science

Before commencing the experiments, we are grateful for the open-source nature of the software and dataset used in this work and have taken into account the benefits that the principles of open science bring to research. Therefore, we have opened our source code, datasets, and pre-trained models on https://zenodo.org/records/14736906, accompanied by a detailed description, *e.g.*, the `README` file. The datasets and models we used are all open-source files, with no proprietary datasets or models, and we have provided references or links to pre-trained models for indexing.

References

[1] Bert-vits2. https://github.com/fishaudio/Bert-VITS2, 2024.

[2] Gpt-sovits. https://github.com/RVC-Boss/GPT-SoVITS, 2024.

[3] Ttsds. https://huggingface.co/spaces/ttsds/benchmark, 2024.

[4] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, and S. Anadkat. Gpt-4 technical report. *arXiv*, 2023.

[5] 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*, 2024.

[6] K. Baba, W. Nakata, Y. Saito, and H. Saruwatari. The t05 system for the VoiceMOS Challenge 2024: Transfer learning from deep image classifier to naturalness MOS prediction of high-quality synthetic speech. In *SLT*, 2024.

[7] J. Betker. Better speech synthesis through scaling. *arXiv*, 2023.

[8] E. Casanova, K. Davis, E. Gölge, G. Göknar, I. Gulea, L. Hart, A. Aljafari, J. Meyer, R. Morais, and S. Olayemi. Xtts: a massively multilingual zero-shot text-to-speech model. In *INTERSPEECH*, 2024.

[9] G. Chen, S. Chenb, L. Fan, X. Du, Z. Zhao, F. Song, and Y. Liu. Who is real bob? adversarial attacks on speaker recognition systems. In *SP*, 2021.

[10] G. Chen and Y. Zhang. Songbsab: A dual prevention approach against singing voice conversion based illegal song covers. In *NDSS*, 2025.

[11] 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 *ICLR*, 2023.

[12] Y. Chen, Z. Niu, Z. Ma, K. Deng, C. Wang, J. Zhao, K. Yu, and X. Chen. F5-tts: A fairytaler that fakes fluent and faithful speech with flow matching. *arXiv*, 2024.

[13] B. Desplanques, J. Thienpondt, and K. Demuynck. Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification. In *INTERSPEECH*, 2020.

[14] F. Fang, X. Wang, J. Yamagishi, I. Echizen, M. Todisco, N. Evans, and J. Bonastre. Speaker anonymization using x-vector and neural waveform models. In *SSW*, 2019.

[15] L. Fowl, M. Goldblum, P. Chiang, J. Geiping, W. Czaja, and T. Goldstein. Adversarial examples make strong poisons. In *NeurIPS*, 2021.

[16] S. Fu, F. He, Y. Liu, L. Shen, and D. Tao. Robust unlearnable examples: Protecting data privacy against adversarial learning. In *ICLR*, 2022.

[17] K. Hashimoto, J. Yamagishi, and I. Echizen. Privacy-preserving sound to degrade automatic speaker verification performance. In *ICASSP*, 2016.

[18] P. He, J. Gao, and W. Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. In *ICLR*, 2023.

[19] C. Huang, Y. Y. Lin, H. Lee, and L. Lee. Defending your voice: Adversarial attack on voice conversion. In *IEEE SLT*, 2021.

[20] H. Huang, X. Ma, S. M. Erfani, J. Bailey, and Y. Wang. Unlearnable examples: Making personal data unexploitable. In *ICLR*, 2021.

[21] S. Hussain, P. Neekhara, S. Dubnov, J. McAuley, and F. Koushanfar. {WaveGuard}: Understanding and mitigating audio adversarial examples. In *USENIX Security*, 2021.

[22] A. Ilyas, L. Engstrom, A. Athalye, and J. Lin. Black-box adversarial attacks with limited queries and information. In *ICML*, 2018.

[23] K. Ito and L. John. The lj speech dataset. https://keithito.com/LJ-Speech-Dataset/, 2017.

[24] W. Jang, D. Lim, J. Yoon, B. Kim, and J. Kim. Univnet: A neural vocoder with multi-resolution spectrogram discriminators for high-fidelity waveform generation. In *INTERSPEECH*, 2021.

[25] Z. Jiang, J. Liu, Y. Ren, J. He, Z. Ye, S. Ji, Q. Yang, C. Zhang, P. Wei, C. Wang, X. Yin, Z. Ma, and Z. Zhao. Mega-TTS 2: Boosting prompting mechanisms for zero-shot speech synthesis. In *ICLR*, 2024.

[26] T. Kaneko, K. Tanaka, H. Kameoka, and S. Seki. istftnet: Fast and lightweight mel-spectrogram vocoder incorporating inverse short-time fourier transform. In *ICASSP*, 2022.

[27] M. Kawamura, Y. Shirahata, R. Yamamoto, and K. Tachibana. Lightweight and high-fidelity end-to-end text-to-speech with multi-band generation and inverse short-time fourier transform. In *ICASSP*, 2023.

[28] J. Kim, S. Kim, J. Kong, and S. Yoon. Glow-tts: A generative flow for text-to-speech via monotonic alignment search. In *NeurIPS*, 2020.

[29] J. Kim, J. Kong, and J. Son. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *ICML*, 2021.

[30] J. Kominek. Cmu arctic databases for speech synthesis. *CMU-LTI*, 2003.

[31] J. Kong, J. Kim, and J. Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. In *NeurIPS*, 2020.

[32] R. Kubichek. Mel-cepstral distance measure for objective speech quality assessment. In *PACRIM*, 1993.

[33] K. Kumar, R. Kumar, T. De Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. De Brebisson, Y. Bengio, and A. C. Courville. Melgan: Generative adversarial networks for conditional waveform synthesis. In *NeurIPS*, 2019.

[34] C. S Legislature. California consumer privacy act of 2018, 2018.

[35] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu. Neural speech synthesis with transformer network. In *AAAI*, 2019.

[36] Y. A. Li, C. Han, V. Raghavan, G. Mischler, and N. Mesgarani. Styletts 2: Towards human-level text-to-speech through style diffusion and adversarial training with large speech language models. In *NeurIPS*, 2024.

[37] S. Liao, Y. Wang, T. Li, Y. Cheng, R. Zhang, R. Zhou, and Y. Xing. Fish-speech: Leveraging large language models for advanced multilingual text-to-speech synthesis. *arXiv*, 2024.

[38] C. Liu, J. Zhang, T. Zhang, X. Yang, W. Zhang, and N. Yu. Detecting voice cloning attacks via timbre watermarking. In *NDSS*, 2024.

[39] Z. Liu, Z. Zhao, and M. Larson. Image shortcut squeezing: Countering perturbative availability poisons with compression. In *ICML*, 2023.

[40] X. Miao, X. Wang, E. Cooper, J. Yamagishi, and N. Tomashenko. Speaker anonymization using orthogonal householder neural network. *TASLP*, 2023.

[41] L. Muñoz-González, B. Biggio, A. Demontis, A. Paudice, V. Wongrassamee, E. C. Lupu, and F. Roli. Towards poisoning of deep learning algorithms with back-gradient optimization. In *AISec*, 2017.

[42] I. Natgunanathan, P. Praitheeshan, L. Gao, Y. Xiang, and L. Pan. Blockchain-based audio watermarking technique for multimedia copyright protection in distribution networks. *TOMM*, 2022.

[43] Y. Ning, S. He, Z. Wu, C. Xing, and L. Zhang. A review of deep learning based speech synthesis. *Applied Sciences*, 2019.

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

[45] Z. Qin, W. Zhao, X. Yu, and X. Sun. Openvoice: Versatile instant voice cloning. *arXiv*, 2023.

[46] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever. Robust speech recognition via large-scale weak supervision. In *ICML*, 2023.

[47] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 2019.

[48] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T. Liu. Fastspeech 2: Fast and high-quality end-to-end text to speech. In *ICLR*, 2021.

[49] S. Rouard, F. Massa, and A. Défossez. Hybrid transformers for music source separation. In *ICASSP*, 2023.

[50] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, and R. Skerrv-Ryan. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In *ICASSP*, 2018.

[51] K. Shen, Z. Ju, X. Tan, E. Liu, Y. Leng, L. He, T. Qin, S. Zhao, and J. Bian. Naturalspeech 2: Latent diffusion models are natural and zero-shot speech and singing synthesizers. In *ICLR*, 2024.

[52] H. Siuzdak. Vocos: Closing the gap between time-domain and fourier-based neural vocoders for high-quality audio synthesis. *arXiv*, 2023.

[53] B. M. L. Srivastava, N. Vauquier, M. Sahidullah, A. Bellet, M. Tommasi, and E. Vincent. Evaluating voice conversion-based privacy protection against informed attackers. In *ICASSP*, 2020.

[54] C. Stupp. Fraudsters used ai to mimic ceo's voice in unusual cybercrime case. *The Wall Street Journal*, 2019.

[55] X. Tan, J. Chen, H. Liu, J. Cong, C. Zhang, Y. Liu, X. Wang, Y. Leng, Y. Yi, and L. He. Naturalspeech: End-to-end text-to-speech synthesis with human-level quality. *TPAMI*, 2024.

[56] V. Tolpegin, S. Truex, M. E. Gursoy, and L. Liu. Data poisoning attacks against federated learning systems. In *ESORIC*, 2020.

[57] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, and F. Azhar. Llama: Open and efficient foundation language models. *arXiv*, 2023.

[58] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, and S. Bhosale. Llama 2: Open foundation and fine-tuned chat models. *arXiv*, 2023.

[59] E. Vickers. The loudness war: Background, speculation, and recommendations. In *Audio Engineering Society Convention 129*, 2010.

[60] D. Wang, M. Xue, B. Li, S. Camtepe, and L. Zhu. Provably unlearnable data examples. In *NDSS*, 2025.

[61] Y. Wang, H. Guo, G. Wang, B. Chen, and Q. Yan. VSMask: Defending against voice synthesis attack via real-time predictive perturbation. In *WISEC*, 2023.

[62] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, and S. Bengio. Tacotron: Towards end-to-end speech synthesis. *arXiv*, 2017.

[63] S. Wu, S. Chen, C. Xie, and X. Huang. One-pixel shortcut: On the learning preference of deep neural networks. In *ICLR*, 2023.

[64] S. Wu, J. Wang, W. Ping, W. Nie, and C. Xiao. Defending against adversarial audio via diffusion model. In *ICLR*, 2023.

[65] C. Yang, Q. Wu, H. Li, and Y. Chen. Generative poisoning attack method against neural networks. *arXiv*, 2017.

[66] J. Yao, Q. Wang, P. Guo, Z. Ning, and L. Xie. Distinctive and natural speaker anonymization via singular value transformation-assisted matrix. *TASLP*, 2024.

[67] D. Yu, H. Zhang, W. Chen, J. Yin, and T. Liu. Availability attacks create shortcuts. In *KDD*, 2022.

[68] Z. Yu, S. Zhai, and N. Zhang. Antifake: Using adversarial audio to prevent unauthorized speech synthesis. In *CCS*, 2023.

[69] 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. *arXiv*, 2019.

[70] H. Zhang, X. Zhang, and G. Gao. Training supervised speech separation system to improve stoi and pesq directly. In *ICASSP*, 2018.

[71] X. Zhang, X. Sun, X. Sun, W. Sun, and S. K. Jha. Robust reversible audio watermarking scheme for telemedicine and privacy protection. *Computers, Materials & Continua*, 2022.

[72] Z. Zhang and P. Huang. Hiddenspeaker: Generate imperceptible unlearnable audios for speaker verification system. In *IJCNN*, 2024.

A Summary and Comparison of Related Work

In Table 8, we describe the comparison between SafeSpeech and related works, including type, whether the perturbation is end-to-end, protection stage, target task, transferability enhancement technique, imperceptibility method, real-time capability, and application scenario. Previous protection techniques can mainly be classified into two categories, *perturbative availability poisons* and *voice protection*. PAP safeguards data by preventing its unauthorized use in model training. "Availability" in the PAP means that the source data cannot be used for training purposes [67]. Voice protection employs adversarial examples to prevent voice cloning at the inference stage and protect personal identification.

B Details of Datasets and Models

In this section, we provide detailed information on datasets and models utilized in the experiments.

Table 9: The detailed information of our selected datasets.

Name	Nums	Speakers (M/F)	Sampling Rate	Max/Min Length	Average Length
D_1	134	1 (0/1)	24000	9.94/0.61 (s)	4.51 (s)
D_2	1800	18 (11/7)	16000	6.69/1.20 (s)	3.15 (s)

D_1 and D_2 represent the dataset LibriTTS and CMU ARCTIC respectively.

Dataset Information. Some details about the two datasets we selected are shown in Table 9 including numbers of samples, speaker categories, sampling rate of the dataset, maximum length, minimum length, and average length of dataset.

Table 10: The detailed information of our selected models.

Models	**Type**	**Structure**	**Vocoder**	**LLM**	**NAR**	**Hours**
BERT-VITS2 [1]	fine-tuning	LLM + VAE	HiFiGAN [31]	DeBERTaV3 [18]	-	-
FishSpeech [37]	zero-shot	LLM + Dual-AR	Firefly-GAN	Llama [57]	✗	720K
F5-TTS [12]		flow matching	Vocos [52]	-	✓	95K
GlowTTS [28]	fine-tuning	flow-based	HiFiGAN [31]			24
MB-iSTFT-V [27]		iSTFT + VAE				24
OpenVoice [45]	zero-shot	encoder-decoder				3.5K
StyleTTS 2 [36]	fine-tuning	diffusion-based	HiFiGAN [31] iSTFTNet [26]			245
TorToise-TTS [7]	zero-shot	diffusion-based	UnivNet [24]		✗	46K
VITS [29]	fine-tuning	VAE	HiFiGAN [31]		✓	24
XTTS [8]	zero-shot	LLM + VQ-VAE		GPT-2 [47]	✗	27K

(1)**VQ-VAE**: Vector Quantized-Variational AutoEncoder. (2) **Dual-AR**: Dual Autoregressive. (3) **iSTFT**: inverse Short-Time Fourier Transform. (4) **FireflyGAN**: FireflyGAN is an enhanced vocoder proposed in the FishSpeech.

Model Information. To facilitate a more comprehensive comparison of the various models, we show the Table 10. This table highlights differences among models across several key dimensions: model type, structure, incorporation of the LLM component, non-autoregressive (NAR) status, and the volume of training dataset for the base models. In cases where models do not incorporate the LLM component, this is indicated with a dash ("-") in the corresponding "LLM" column. It should be noted that for BERT-VITS2, official disclosure regarding the volume of training data is unavailable.

C Details of User Study

Filtering. To ensure participant seriousness, we designed two simple English arithmetic questions in the questionnaire at random positions. Incorrect answers to these questions indicated a lack of seriousness, leading to the exclusion of those responses. Additionally, we filtered out participants who provided identical or random answers throughout.

Experimental Generalizability. To minimize the bias introduced by subjective experiments and enhance the generalizability of the survey, we have employed several techniques. (1) *Adequate Participants*. Compared to related works such as AntiFake [68], which used 24 participants, and VSMask [61], which used 25 participants, we invited 80 individuals to take part in our survey. (2) *Randomization and Anonymization*: The order of questions in the questionnaire was completely randomized. We provided no hints or additional instructions within the questions, and participants had no way of knowing which algorithm generated the audio they were currently listening to. (3) *Confidence Interval*: Considering the potential bias, *e.g.*, personal preferences of participants, in result calculations of the subjective survey, we computed a 95% confidence interval for MOS and naturalness values to provide more reliable results as Eq. (11) and (12).

We assume that the MOS score of model i is μ_i, and in addition, the 95% confidence interval [33] score is CI_i, which can be calculated using the following formula:

$$\hat{\mu}_i = \frac{1}{N_i} \cdot \sum_{k=1}^{N_i} m_{i,k}, \tag{11}$$

$$CI_i = \left[\hat{\mu}_i - 1.96 \frac{\hat{\sigma}_i}{\sqrt{N_i}}, \hat{\mu}_i + 1.96 \frac{\hat{\sigma}_i}{\sqrt{N_i}} \right], \tag{12}$$

where $\hat{\sigma}_i$ is the standard deviation of the scores collected.

Rating Principles In the questionnaire, participants listened to each audio and rated its quality from 0 to 5 based on their subjective perception. A score of 5 indicates excellent audio quality (smooth and noise-free). 4 suggests good quality (minimal noise and delays, easy to understand). 3 means average quality (with noise and delays but understandable). 2 denotes fairly poor quality (requiring multiple repetitions to understand). 1 indicates poor quality (very hard to understand), and 0 represents extremely poor quality (completely inaudible).

D Additional Experiments

In this section, we conduct an easier fine-tuning model, leveraging a web interface operation. Additionally, we study the impact of the surrogate model and noise radius of the ℓ_p norm.

Table 8: Comparison of SafeSpeech and related works.

Method	Type	E2E	Stage	Target Task	Transferability	Imperceptibility	Real-Time	Application Scenario
Unlearnable Examples [20]	Perturbative Availability Poison	✓	Training	image classification	✗	ℓ_∞ norm	✗	Protect data against authorized training.
AdvPoison [15]					checkpoint ensemble			
SEP [11]								
PTA [20, 29]		✗			✗	patch segment		
AttackVC [19]	Voice Protection of Identification		Inference	voice conversion		ℓ_∞ norm		Protect personal voice information against malicious voice cloning.
VSMask [61]		✓		zero-shot speech synthesis			✓	
AntiFake [68]					encoder ensemble	Frequency Penalty and SNR	✗	
SafeSpeech (ours)	Voice Protection of Synthesis Quality and Identification	✓	Training	zero-shot and fine-tuning speech synthesis	pivotal and universal objective optimization	STFT and STOI metrics (time and frequency domain)	✓	

(1)**Transferability**: The improvement techniques of transferability. (2) **E2E**: "End-to-End" represents whether the perturbation is embedded into the entire waveform not in the latent space. (3) The source code of VSMask is not public and unavailable.

Table 11: The transferability performance when regarding MB-iSTFT-VITS as our surrogate model.

Method	BERT-VITS2			MB-iSTFT-VITS		
	MCD(↓)	WER(%)(↓)	SIM(↑)	MCD(↓)	WER(%)(↓)	SIM(↑)
clean	5.099	25.095	0.625	5.139	20.913	0.623
PTA	9.949	59.646	0.266	8.892	47.569	0.219
SPEC (ours)	**12.791**	**93.552**	**0.215**	**12.374**	**124.142**	**0.159**

Table 12: Quantitative analysis on different perturbation boundary ε with 4/255 and 16/255 on BERT-VITS2 model.

Methods	4/255			16/255		
	MCD(↑)	WER(%)(↑)	SIM(↓)	MCD(↑)	WER(%)(↑)	SIM(↓)
clean	5.099	25.095	0.625	5.099	25.095	0.625
random noise	5.334	34.575	0.584	6.614	43.142	0.433
AdvPoison [15]	7.059	46.030	0.403	13.718	91.525	0.205
SEP [11]	6.084	44.926	0.401	12.737	83.804	0.263
PTA [20]	7.360	49.553	0.342	17.040	89.875	0.206
SPEC (ours)	**10.921**	**76.110**	**0.302**	**18.634**	**105.385**	**0.093**

D.1 WebUI Operation

Recently, an efficiently fine-tuning TTS synthesis model named GPT-SoVITS [2] has garnered over **38K** stars in the GitHub community, which supports WebUI-based training.

GPT-SoVITS has received numerous positive feedback as public users are amazed at its convenience, high performance, and efficient training operation. It consists of GPT [47] and a VITS-based synthesizer, utilizing an LLM component to make the synthesizer understand the text better. The two parts are trained separately. To reproduce the attackers' training process, we use the web-based training method provided by the authors. We upload SafeSpeech-protected audio to the designated path and do audio size division, automatic text recognition, and annotation. Unlike previous experiments where training text was manually annotated, we employ Whisper in Section 5.4 to recognize text. Then we set the training iterations for SoVITS and GPT as 25 and 50 respectively and evaluate the performance on the web-based page.

The result shows that the SIM metric of the synthetic speeches is only 0.25, meaning the dissimilarity between synthetic and original speeches. In this experiment, we fine-tune a well-known and advanced TTS model by web-based operation, which ensures that we have not modified the model and simulate a possible training scenario in the real world.

D.2 Alternative Surrogate Model Choice

SafeSpeech protects datasets based on the surrogate model. In previous experiments, we choose BERT-VITS2 as the surrogate model in our previous experiments, while our methods have no limitation on model selection. Therefore, the user can utilize the specific model for specific scenarios, *e.g.*, MB-iSTFT-VITS in real-time applications.

Table 11 shows the experimental results when regarding MB-iSTFT-VITS as the surrogate model. We can find that after training on the protected dataset, the value WER of 124.142% means highly unclear synthesized audio, with a speaker similarity from 0.623 trained on the clean samples to 0.159, and the ASR of the speech synthesis attack is only 3.846% representing a successful defense. Moreover, the perturbation is also transferable on BERT-VITS2. This experiment serves as an excellent demonstration of the versatility of SafeSpeech in design, that noise generation does not depend on a specific model with high effectiveness and transferability.

D.3 Perturbation Boundaries

The perturbation radius ε plays an important role in the ℓ_p norm constraints of SafeSpeech. Higher ε can achieve better effects but worse perception. Table 12 presents the results when ε is 4/255 and 16/255, respectively. When ε takes 16/255, it can be found that the data protection effect is excellent, with SIM of only 0.093 and attack success rate of 0%, and WER of 105.385%, which means a successful defense. If users aim to realize stronger strength to mitigate the circumvention of SafeSpeech, larger ε can be set, containing the perceptual optimization in Section 4.2.