E2E-VGuard: Adversarial Prevention for Production LLM-based End-To-End Speech Synthesis

Zhisheng Zhang[1,2], **Derui Wang**[3], **Yifan Mi**[2], **Zhiyong Wu**[1*],
Jie Gao[1], **Yuxin Cao**[5], **Kai Ye**[6], **Minhui Xue**[3,4], **Jie Hao**[2*]
[1] Shenzhen International Graduate School, Tsinghua University
[2] Beijing University of Posts and Telecommunications [3] CSIRO's Data61
[4] Responsible AI Research (RAIR) Centre, The University of Adelaide
[5] National University of Singapore [6] The University of Hong Kong
[*] Corresponding authors

Abstract

Recent advancements in speech synthesis technology have enriched our daily lives, with high-quality and human-like audio widely adopted across real-world applications. However, malicious exploitation like voice-cloning fraud poses severe security risks. Existing defense techniques struggle to address the production large language model (LLM)-based speech synthesis. While previous studies have considered the protection for fine-tuning synthesizers, they assume manually annotated transcripts. Given the labor intensity of manual annotation, end-to-end (E2E) systems leveraging automatic speech recognition (ASR) to generate transcripts are becoming increasingly prevalent, *e.g.*, voice cloning via commercial APIs. Therefore, this E2E speech synthesis also requires new security mechanisms. To tackle these challenges, we propose E2E-VGuard, a proactive defense framework for two emerging threats: (1) production LLM-based speech synthesis, and (2) the novel attack arising from ASR-driven E2E scenarios. Specifically, we employ the encoder ensemble with a feature extractor to protect timbre, while ASR-targeted adversarial examples disrupt pronunciation. Moreover, we incorporate the psychoacoustic model to ensure perturbative imperceptibility. For a comprehensive evaluation, we test 16 open-source synthesizers and 3 commercial APIs across Chinese and English datasets, confirming E2E-VGuard's effectiveness in timbre and pronunciation protection. Real-world deployment validation is also conducted. Our code and demo page are available at https://wxzyd123.github.io/e2e-vguard/.

1 Introduction

High-quality synthetic speech based on deepfake techniques [1, 2] has been applied in daily scenarios, such as video dubbing, and vehicle-mounted voice assistants. Large language models (LLMs) [3, 4] have furthered the development of speech synthesis, *i.e.*, text-to-speech (TTS). Current TTS models enhance the synthesis performance by integrating LLMs as a core component for paralanguage features, achieving human-level results. The most advanced technique can be based on an audio foundation model [5]. TTS models can be divided into two categories, *i.e.*, zero-shot [1] and fine-tuning-based [6]. Zero-shot models utilize reference audio as the prompt to clone the voice. In contrast, fine-tuning-based models require a few minutes of speech samples to replicate the target speaker better. The advancements in speech synthesis, on the one hand, bring huge convenience; on the other hand, they pose a potential security threat in the hands of pirate users. These pirate users may conduct illegal speech synthesis for illegal purposes, such as telecommunication fraud. Therefore, the prevention approach against unauthorized synthesis is of vital importance.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Existing Defenses. Existing protective methods against voice cloning focus on two main types: (1) Defense based on adversarial examples (AEs), *e.g.*, AntiFake [7], and AttackVC [8]. (2) Defense based on unlearnable examples (UEs), *e.g.*, POP [9], and SafeSpeech [10]. AEs-based protection generates audio AEs through TTS models' encoders to prevent zero-shot voice cloning. In contrast, UEs-based protection utilizes a universal objective to generate model-agnostic perturbation, which disrupts the training phase of TTS models to achieve fine-tuning-based speech synthesis.

Limitations and Challenges. Prior studies have some limitations when considering broader scenarios: (1) *Industrial-level and LLM-based TTS*. LLMs have advanced the previous deep neural network (DNN)-based speech synthesis. Common approaches employ a speech tokenizer to encode the input waveform into discrete tokens, which are fed into LLMs. The decoded outputs from LLMs then guide the generation module, such as the flow-matching module [1]. The key distinction lies in decoding audio signals into discrete tokens rather than continuous embeddings, a direction rarely explored in this protection of LLM-based TTS research. Moreover, voice replication products have emerged in the industry, where voice cloning via API constitutes the focus of this paper. (2) *End-To-End Scenario*. The assumption in prior studies is that the text corresponding to the audio has been provided. A more realistic scenario is that, for a customized dataset, the text needs to be obtained by the training party itself. For example, the text transcripts are transcribed through an automatic speech recognition (ASR) system, rather than relying on manually annotated open-source datasets. In fact, commercial APIs typically accept only audio input and rely on an ASR on their backend.

Why is End-to-End Fine-Tuning Important? End-to-end fine-tuning aims to ensure that both input and output data are exclusively of audio type and integrates an ASR system into the training process for text recognition. Based on this, two research questions (RQ) should be considered. **RQ1:** *Why fine-tuning?* On the one hand, some models, such as VITS [6], only support fine-tuning without zero-shot capabilities. On the other hand, fine-tuning can achieve better synthetic performance than relying solely on a single sample in zero-shot scenarios. **RQ2:** *Why end-to-end fine-tuning?* Adversaries often collect audio data from public social platforms like YouTube and Bilibili, where the audio does not come with corresponding text data. Manually annotating text is typically time-consuming and labor-intensive. Therefore, leveraging an ASR system based on deep learning techniques is a more practical choice due to its efficiency and high recognition accuracy. In the real world, the industrial product for speech synthesis trains a new speaker via an API connection with only audio input. The supplemental ASR system is utilized for automatic recognition in their service.

Our Solution and Contributions. To counter the LLM-based and end-to-end scenarios, we propose *E2E-VGuard*, a proactive defense framework that disrupts both timbre and pronunciation. For the timbre, we introduce untargeted and targeted speaker protection based on the proposed feature loss, which utilizes ensembled encoders and a feature extractor to obtain audio features for LLM-based TTS, resulting in dissimilar synthetic speeches to achieve identification protection. For the pronunciation, E2E-VGuard generates the audio AEs to fool the ASR system with incorrectly recognized text and disrupt model's learning process of the text and pronunciation. Moreover, to realize imperceptibility, we introduce the psychoacoustic model [11] to add the perturbation within a specific frequency domain, reducing the detection by the human ears. For a comprehensive evaluation, we conduct experiments on both English and Chinese datasets, verifying their effectiveness and transferability across 16 open-source, 3 commercial models, and 7 ASR systems. E2E-VGuard is robust against sophisticated data augmentation and perturbation removal techniques. Moreover, we have validated the E2E-VGuard's robustness in the real world. Our contributions can be summarized as follows:

- We introduce a more realistic and challenging scenario of end-to-end fine-tuning-based speech synthesis, and we propose a proactive framework, E2E-VGuard, to protect individual information.
- We consider defensive waveform disruption from the perspectives of timbre and pronunciation. For the timbre disruption, we propose a feature objective based on the encoder ensemble and feature extractor. For the pronunciation disruption, we utilize AEs against ASR systems to fool TTS models with incorrect text and impact pronunciation.
- We utilize the psychoacoustic model with ℓ_2-norm to enhance the perturbation imperceptibility for better human audible perception.
- We evaluate the effectiveness, transferability, and robustness of E2E-VGuard through comprehensive experiments across diverse settings: 19 TTS models (including 16 open-source and 3 commercial), 7 ASR systems, and 3 English and Chinese datasets.

2 Related Work

Speech Synthesis. Modern speech synthesis can be primarily categorized into two types: DNN-based and LLM-based architectures. The former mainly focuses on building a synthesizer [6, 12], while the latter integrates the LLM with a synthesizer and encodes audio into discretized tokens [1]. The LLM captures the prosody and semantic features, while the synthesizer captures timbre and environmental information [1]. Recently, *audio foundation models* have advanced rapidly. During the pre-training phase, models typically acquire basic question-answering capabilities. In the post-training phase, they gain downstream tasks, *e.g.*, TTS, through supervised fine-tuning (SFT). This emerging TTS approach based on speech foundation models is also the focus of this paper, *e.g.*, Step-Audio [5].

Voice Protection. Defenses against synthetic speech can be categorized into proactive and passive defenses [7, 13, 14]. We primarily focus on proactive defense techniques, which aim to reduce speaker similarity in synthesized audio at the data source. For example, Huang *et al.* [8] and Yu *et al.* [7] utilized adversarial examples to disrupt voice cloning. Recently, Zhang *et al.* [9, 10] proposed unlearnable samples to degrade the quality of speech synthesis systems, thereby defending against fine-tuning-based voice synthesis. In addition, we note recent advances in passive defense techniques, such as the robust deepfake audio detection method proposed by Zhang *et al.* [15], which aims to mitigate the risks posed by synthetic audio. However, existing approaches remain ineffective against more *advanced TTS models and end-to-end scenarios at the data level*.

3 E2E-VGuard Design

3.1 Threat Model

In this section, we analyze the necessity of end-to-end fine-tuning through two examples.

ByteDance's API.[1] Taking ByteDance's voice cloning product as an example, third-party users upload audio to the company's server infrastructure via the API. First, the input audio is transcribed into text using an ASR system. Then, both the transcribed text and audio information are fed into the TTS model for voice training, enabling synthesis of target text using the trained voice timbre. This commercial API-based speech synthesis approach also fits into the end-to-end scenario.

Open-sourced WebUI Operation. GPT-SoVITS [2] is an open-source zero-shot voice cloning model and supports few-sample fine-tuning to improve voice similarity. The project provides a WebUI-based fine-tuning workflow: the corresponding text is first obtained utilizing an ASR system, *i.e.*, Whisper by OpenAI [16]. Subsequently, fine-tuning is performed based on the text and audio, which also satisfies the end-to-end scenario requirements.

From these two examples, we can observe that end-to-end speech synthesis is prevalent in real-world applications. Moreover, ASR-based workflows are gradually becoming mainstream, which typically run in model or server backends and are thus not directly accessible to end-users.

3.2 Problem Formulation

From the perspectives of timbre and pronunciation, we implement E2E-VGuard for voice anti-cloning. At the timbre level, previous work has focused on targeted [8, 7] and untargeted [9] attacks, *i.e.*, whether to select a specific target speaker for timbre perturbation. We consider two approaches to timbre protection with a broader defensive selection: targeted and untargeted. For pronunciation protection, adversarial targeted attacks are employed against ASR systems, ensuring that the ASR system transcribes into a specific target text. This is because we aim for the transcribed text to be meaningful rather than gibberish, thereby reducing the adversary's detection of textual alterations. The workflow is shown in Figure 1, and the objective function of E2E-VGuard can be expressed as:

$$\begin{aligned} &\mathcal{L}(x') = \mathcal{L}_{asr}(x') + \alpha \cdot \mathcal{L}_{fea}(x') + \beta \cdot \mathcal{L}_{psy}(x'), \\ &\text{s.t.} \quad ||x' - x||_p \leq \epsilon \text{ and } x' \in [-1, 1]^T, \end{aligned} \tag{1}$$

where $\mathcal{L}_{asr}(\cdot)$ represents the loss of the ASR system, $\mathcal{L}_{fea}(\cdot)$ denotes the speech feature loss, and $\mathcal{L}_{psy}(\cdot)$ is the perceptual optimization function for embedded perturbations, *i.e.*, the psychoacoustic

[1] https://www.volcengine.com/docs/6561/133350

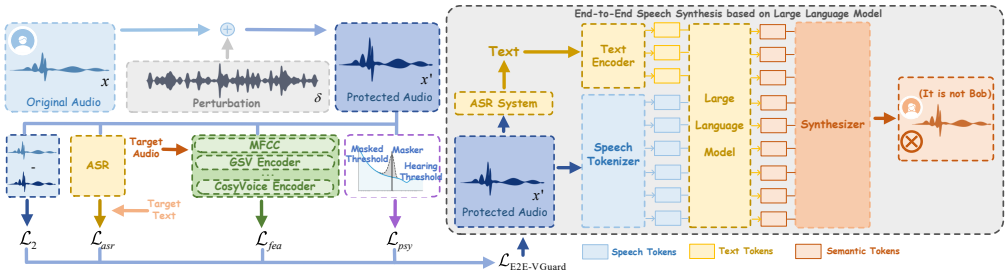


Figure 1: The workflow of E2E-VGuard and the end-to-end speech synthesis pipeline with LLM.

model. α and β are weight coefficients for multi-task balance. Moreover, x and x' represent the original and protected audio, respectively, with features normalized to fall within the range $[-1, 1]$. The perturbation $\delta := x' - x$ is bounded by an ℓ_p-norm constraint ϵ. T is the waveform length.

3.3 Timbre Prevention

TTS models typically generate high-quality timbre-similar audio using reference speech samples. We implement timbre-level voice protection to achieve anti-cloning voice protection and ensure cloned audio dissimilarity from the original speakers. Previous research [7, 8] primarily employs encoders for timbre extraction and optimization to create timbre divergence between original and target audio. Building upon the timbre encoder ensemble [7], we further incorporate acoustic features by the MFCC [17] extractor to better perturb speaker identity features. In designing E2E-VGuard, we propose two protection methods: targeted and untargeted timbre protection.

Untargeted Timbre Protection. Untargeted protection maximizes feature distance between original audio x and protected audio x', rendering synthesized audio unrecognizable as the original speaker. Zero-shot TTS models typically employ an encoder to extract cloning-relevant features, *e.g.*, timbre in CosyVoice [1] and style in StyleTTS2 [12], combining them with pre-trained articulation patterns for speech synthesis. Building on prior findings demonstrating enhanced transferability through encoder ensemble [7, 18], we extract timbre and acoustic features through multiple encoders from target TTS models to improve the protective generalizability against unseen models. Moreover, to counter LLM-based TTS, we consider to protect at the audio's original features by changing the discrete tokens obtained by the audio tokenizer for the LLM component with the MFCC extractor to protect articulation and prosodic patterns. The objective function can be formulated as:

$$\mathcal{L}_{fea}(x') = \sum_{i=1}^{k} \mathrm{CS}(E_i(x), E_i(x')) + \mathrm{CS}(M(x), M(x')), \tag{2}$$

where k is the number of selected encoders, $\mathrm{CS}(\cdot,\cdot)$ is cosine similarity with lower values indicating reduced similarity [18], $E(\cdot)$ is the timbre encoder, and $M(\cdot)$ denotes the MFCC extractor.

Targeted Timbre Protection. Targeted protection steers original audio features toward a designated target speaker x_t, causing TTS models to synthesize target-like audio. For target selection, we construct a speaker database following AntiFake [7], choosing the most dissimilar speaker by feature distance for each protected audio. This systematically identifies the most dissimilar speaker in feature space, contrasting with conventional random opposite-gender sampling. The optimization function is:

$$\mathcal{L}_{fea}(x') = -\sum_{i=1}^{k} \mathrm{CS}(E_i(x_t), E_i(x')) - \mathrm{CS}(M(x_t), M(x')). \tag{3}$$

Both methods can effectively protect the speaker's identity. Untargeted protection enables broader adversarial sample exploration due to undefined optimization targets. Targeted protection leverages carefully selected feature-divergent samples for enhanced timbre disruption.

3.4 Pronunciation Prevention

Fine-tuning a TTS model requires pairs of text and audio data to achieve alignment between and pronunciation. For instance, VITS utilizes the monotonic alignment search algorithm to search for

the correspondence between time frames and characters. Text data can be obtained through manual annotation or automatic recognition by an ASR system. The former consumes a lot of manpower, time, and costs, therefore, it is more common to employ an ASR system to recognize text information. Previous work [19, 20, 21] has shown that ASR systems are relatively vulnerable and susceptible to adversarial examples that interfere with recognition accuracy. Based on this finding, we consider utilizing adversarial examples to disrupt the ASR system's recognition, causing the protected audio to be recognized as a different text. Incorrect text-audio pairs will disrupt the pre-trained model's learning of the pronunciation, preventing the synthesis of audio with the corresponding pronunciation based on the desired text, thereby effectively protecting personal unauthorized audio data.

Adversarial attacks on ASR systems can also be divided into targeted and untargeted types, differing in whether a specific target text is provided when optimizing. The generated text by the untargeted attack is incoherent, while targeted attacks ensure the readability of recognized text by specifying the text, effectively reducing the adversary's awareness of the anomalous recognition text. Therefore, we choose targeted attacks against ASR systems. The selection of the target text affects the effectiveness of adversarial examples. For instance, long audio paired with short target text may result in the latter part of the recognized text retaining the original correct text, necessitating further optimization considerations. Therefore, we need to consider the selected text and its length. In targeted attacks on timbre in Section 3.3, since the chosen audio already contains specific pronunciation information different from the original audio, and the application of MFCC also benefits the ASR system's recognition of the target audio text [19] in the optimization of adversarial examples, we select the target audio's text as the target text. For untargeted attacks on timbre, we select text of the same length as the audio for different audio, which is more beneficial for adversarial attacks against ASR systems. In summary, the perturbation of pronunciation information can be represented as:

$$\mathcal{L}_{asr}(x') = \mathcal{F}\left(\mathrm{ASR}(x'), y_t\right), \tag{4}$$

where $\mathcal{F}$ is the objective function of the ASR system, such as the connectionist temporal classification (CTC) [22] loss for Wav2vec2 [23]. $\mathrm{ASR}(\cdot)$ computes the input audio to obtain outputs, such as the probability distribution of recognized words. Additionally, y_t denotes the targeted text.

By optimizing Eq. (4), the adversary utilizes the ASR system and obtains incorrect text, thereby interfering with text-pronunciation alignment, making the synthesized audio unintelligible.

3.5 Psychoacoustic Model

The perturbation embedded in a specific region will be masked, making it imperceptible for the human ear to hear the sound in that region, which is known as the *masking effect*. Leveraging this characteristic, we optimize the perception of the embedded perturbation utilizing the psychoacoustic model to enhance the naturalness and imperceptibility. The *masking effect* can be divided into *temporal masking* and *frequency masking*. Following the settings of V-Cloak [24], we employ the *frequency masking* part to ensure perturbations are imperceptible to the human ear.

We set the original audio as the *masker* so that the perturbation *(maskee)* remains below the masking threshold. Let F represent the total number of frequencies and θ_x represent the masking threshold of the original audio x, with each element indicating the maximum acceptable perturbation at frequency f. Assume p_x represents the log-magnitude power spectral density (PSD) of audio x. Therefore, the objective function of the psychoacoustic model can be expressed as:

$$\mathcal{L}_{psy}(x') = \frac{1}{F}\sum_{f=1}^{F}\max\left(0, p_{x'-x}(f) - \theta_x(f)\right), \tag{5}$$

where $\max(0, \cdot)$ ensures the value is non-negative.

Additionally, we impose constraints on the perturbation through ℓ_2, as Duan *et al.* [25] found that ℓ_2 performs optimally in correlation with human perception within the ℓ_p norm. Therefore, we further reduce the perceptibility of the embedded perturbations by introducing $\mathcal{L}_2$, formulated as:

$$\mathcal{L}_2(x') = ||x' - x||_2. \tag{6}$$

To ensure that the protected audio does not exceed the range it should belong to, after obtaining the final protected audio, we map its range back to between -1 and 1 to guarantee it is a normal audio waveform. In conclusion, the algorithm of E2E-VGuard has been provided in Appendix B.

4 Experiments and Analyses

Experiment Organization. In this section, we introduce our experimental evaluation. We first provide our experimental settings in Section 4.1. Then, we evaluate the effectiveness of end-to-end fine-tuning-based speech synthesis in Section 4.2, zero-shot scenarios in Section 4.3, and commercial API test in Section 4.4. Moreover, we explore the effect of each component in Section 4.5 and test the robustness of E2E-VGuard in Section 4.6. In the Appendix, there is also an inevitable evaluation. We validate the effect across multilingual and multi-speaker settings in Appendix F and various ASR systems in Appendix E. Finally, we conduct a human survey of the effectiveness and perception in Appendix G. All of our experiments are conducted on one NVIDIA 4090 GPU. Moreover, the ethical considerations about human study and commercial test are provided in the Appendix, and some limitations and discussions of the E2E-VGuard have been discussed in Appendix A.

4.1 Experimental Settings

In this section, we introduce the experimental settings utilized in our experiments.

Synthesizers. We select a total of 16 TTS models for evaluation. Section 4.2 includes 6 models: GPT-SoVITS (GSV) [2], CosyVoice [1], Llasa-1B [26], Llasa-8B [26], StyleTTS2 [12], and VITS [6], used for end-to-end fine-tuning tests. Section 4.3 includes 7 models: Index-TTS [27], FireRedTTS-1S [28], Step-Audio-TTS [5], Spark-TTS [29], XTTS [30], FishSpeech [31], and Dia-1.6B [32], for zero-shot validation. Moreover, we test 3 models based on in-context learning rather than speaker encoder for feature extraction in Section 4.3: VALLE-X [33], E2-TTS [34], and F5-TTS [35]. Section 4.4 involves 3 commercial APIs: ByteDance, Alibaba, and MiniMax. Notably, Step-Audio-TTS is developed through post-training of a speech foundation model. Further details about open-source models' architectures and sources have been provided in the Appendix D.1.

Encoders. In Section 3.3, six encoders serve as feature extractors: posterior encoders from VITS and GSV, MFCC features, WavLM [36], CAM++ [37] from CosyVoice, and the style encoder from StyleTTS2. Among these, MFCC represents traditional acoustic features, WavLM is a speaker verification system, while the remaining four are timbre or style encoders from TTS systems. This multi-encoder framework improves E2E-VGuard's cross-model transferability in timbre preservation.

ASR Systems. Adversaries may employ different ASR systems to recognize text. We conduct model-specific adversarial attacks against ASR systems and can effectively induce misclassification. Seven ASR models are selected, namely Wav2vec2 [23], Whisper (base, small, medium, and large) [16], Conformer [38], and CitriNet [39]. Appendix D.2 shows different structures in detail.

Datasets. We selected both single-speaker and multi-speaker datasets in English and Chinese to verify E2E-VGuard's protection performance across different scenarios, employing LibriTTS [40] for English single-speaker evaluation following [10], CMU ARCTIC [41] for English multi-speaker testing, and THCHS30 [42] for Chinese multi-speaker assessment. For each dataset, we have randomly allocated 80% for training and 20% for testing. If the model requires a validation set, we utilize 10% of the training set as the validation set.

Metrics. The strength and perception metrics are considered.

- *Word Error Rate* (WER) [9]. It represents the speech intelligibility. *Higher WER reflects lower speech quality*. We utilize a pre-trained ASR model, OpenAI's Whisper with medium size [16].
- *Speaker Similarity* (SIM) [10]. SIM measures the speaker similarity of two speeches. *Lower SIM reflects lower similarity between original and synthetic speeches*. We employ ECAPA-TDNN [43] to extract speaker embeddings and compute SIM values.
- *Signal-to-Noise Ratio* (SNR) [10]. SNR reflects the ratio of the embedded perturbation.
- *Perceptual Evaluation of Speech Quality* (PESQ) [18]: PESQ is an objective perceptual score of the speech quality, ranging from -0.5 to 4.5.
- *Mean Opinion Score* (MOS) [9]. MOS is obtained through human interaction as a subjective metric, which measures the human perception of speeches, ranging from 0 to 5.

Hyperparameter Settings. For fine-tuning, we keep the conventional settings with training details in Appendix D.1. Moreover, the hyperparameters in Eq. (1) are set to balance the effectiveness and

Table 1: Effectiveness and Perception Results of End-To-End Fine-tuning-based Speech Synthesis. The best and second-best protective results are highlighted with **bold** and underlined, respectively.

Method	GSV [2]		CosyVoice [1]		Llasa-1B [26]		Llasa-8B [26]		StyleTTS2 [12]		VITS [6]		Imperceptibility	
	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	SNR(↑)	PESQ(↑)
clean	3.434	0.685	4.288	0.700	3.157	0.643	7.449	0.643	1.895	0.731	7.796	0.710	-	-
AttackVC [8]	5.205	0.636	5.531	0.688	15.201	0.569	9.800	0.593	2.056	0.674	9.039	0.631	-2.456	**3.890**
AntiFake [7]	28.846	0.149	7.841	0.232	22.391	0.250	15.500	0.284	3.623	0.283	41.491	0.257	12.839	1.759
POP [9]	3.573	0.671	4.452	0.715	7.283	0.684	5.692	0.675	1.756	0.743	13.281	0.685	18.425	3.318
POP+ESP [9]	40.308	0.268	10.312	0.259	27.343	0.280	34.639	0.297	7.770	0.298	55.811	0.149	11.246	1.671
SafeSpeech [10]	44.777	0.339	8.596	0.459	9.367	0.288	16.970	0.269	5.215	0.366	105.524	0.180	7.647	1.412
E2E-VGuard (UT)	66.471	**0.123**	21.566	**0.091**	**74.956**	**0.155**	80.221	**0.134**	45.836	**0.082**	95.740	**0.106**	18.523	1.949
E2E-VGuard (T)	**94.812**	0.284	**72.143**	0.375	63.945	0.442	**89.510**	0.310	**54.732**	0.229	**125.299**	0.245	**20.470**	2.324

Table 2: Protective performance across industrial-level and LLM-based models of E2E-VGuard under zero-shot end-to-end speech synthesis.

Method	Index-TTS [27]		FireRedTTS-1S [28]		Step-Audio-TTS [5]		Spark-TTS [29]		XTTS-v2 [30]		FishSpeech [31]		Dia-1.6B [32]	
	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)
clean	3.547	0.674	1.382	0.655	2.508	0.579	1.341	0.666	1.258	0.555	1.848	0.497	4.526	0.581
AntiFake [7]	3.685	**0.147**	3.152	0.276	2.395	0.206	4.214	0.243	7.115	0.193	8.495	0.178	5.471	0.283
POP+ESP [9]	3.730	0.282	4.491	0.366	5.845	0.312	30.667	0.180	7.059	0.358	6.998	0.129	6.266	0.315
SafeSpeech [10]	**5.532**	0.244	6.098	0.339	5.764	0.334	23.866	0.144	**13.522**	0.230	**21.335**	0.197	58.669	0.250
E2E-VGuard (UT)	4.474	0.196	6.528	**0.226**	**8.156**	**0.008**	33.357	**0.174**	5.761	**0.173**	9.746	**0.127**	**83.200**	**0.208**
E2E-VGuard (T)	2.667	0.441	**40.338**	0.367	3.245	0.128	**72.522**	0.260	3.014	0.455	7.633	0.218	35.811	0.248

imperceptibility of each component. We determine hyperparameters through experiments evaluating both loss values and component effectiveness, ultimately selecting $\alpha = 500$ and $\beta = 5 \times 10^{-3}$. Additionally, the ϵ in Eq. (1) is $8/255$, and we optimize perturbation for 500 iterations.

4.2 Effectiveness on End-To-End Fine-Tuning Scenarios

To assess the effectiveness and transferability, we utilize E2E-VGuard to protect the LibriTTS dataset with the untargeted and targeted mode in Section 3.3 and set the target ASR system as Wav2vec2 [23].

Fine-tuning on Protected Dataset. After protecting the LibriTTS dataset, users can upload it publicly to social platforms. Adversaries may require these samples unauthorizedly and utilize advanced synthesizers for fine-tuning-based speech synthesis.

Speech Synthesis and Evaluation. TTS models possess the capabilities of speech synthesis after fine-tuning. Fine-tuning-based models can generate speeches with speaker ID and synthesized text, while zero-shot models require reference audio and synthesized text for feature extraction and cloning. Table 1 shows the experimental results across different TTS models after fine-tuning on datasets protected by different strategies. We can find that our protected E2E-VGuard can achieve an outstanding protective strength than baselines in terms of timbre (SIM) and pronunciation (WER). For fine-tuning-based models, the E2E-VGuard achieves an average increase of 19.775% (targeted, T) in WER compared to the best baseline values while reducing SIM by an average of 0.043 (UT), indicating lower speech intelligibility and similarity. The protection effect improves more significantly on zero-shot models, with WER increasing by an average of 32.841% (UT) and 50.060% (T) and SIM decreasing by 0.119 (UT). It demonstrates that audio protected by the E2E-VGuard effectively safeguards private information, prevents high-quality speech synthesis, and exhibits strong transferability across models.

Perception Analyses. The embedded perturbation should not interfere with the normal utilization of speeches. Table 1 presents the simulated perception metrics of different baselines. The SNR values are higher than all baselines, representing the lowest noise ratio and quality disruption.

4.3 Effectiveness on End-To-End Zero-shot Scenarios

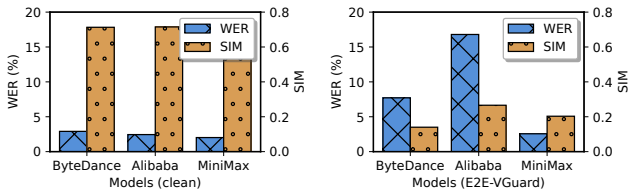
In this section, we conduct zero-shot end-to-end speech synthesis on seven industrial-level and LLM-based TTS models. The reference audio transcripts are automatically obtained through an ASR system. Table 2 presents the test results, where our E2E-VGuard (UT) achieves SOTA performance in voice timbre preservation across all models than baselines. Regarding pronunciation prevention, the average WER values of E2E-VGuard are 21.603% for UT and 23.604% for T, respectively, outperforming AntiFake's 4.932% and SafeSpeech's 19.255%. This indicates the synthesized audio demonstrates both dissimilar timbre characteristics and reduced pronunciation clarity.

Table 3: Evaluation on ICL-based TTS models.

Method	F5-TTS [35]		E2-TTS [34]		VALLE-X [33]	
	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)
clean	4.268	0.676	5.401	0.678	14.450	0.519
AntiFake [7]	4.303	0.282	4.004	0.269	96.469	0.249
E2E-VGuard (UT)	10.776	**0.053**	7.064	**0.138**	**129.483**	**0.175**
E2E-VGuard (T)	**70.034**	0.319	**84.913**	0.372	88.707	0.176

Table 4: The ablation study of the E2E-VGuard.

w/ o	Effectiveness				Imperceptibility	
	VITS		GSV			
	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	SNR(↑)	PESQ(↑)
$\mathcal{L}_{psy}$ & $\mathcal{L}_2$	**119.242**	**0.101**	**90.436**	**0.081**	12.942	1.544
$\mathcal{L}_{fea}$	101.407	0.177	65.446	0.409	15.065	1.811
$\mathcal{L}_{asr}$	94.940	0.102	49.059	0.124	13.724	1.572
E2E-VGuard (UT)	95.740	0.106	66.471	0.123	**18.523**	**1.949**



(a) Clean mode. (b) E2E-VGuard mode.

Figure 2: Evaluation of protective performance on commercial APIs.

The experimental results from Section 4.2 and Section 4.3 demonstrate that our proposed E2E-VGuard effectively protects the latest open-source industrial-level and LLM-based models from timbre and pronunciation perspectives. The method achieves current SOTA performance levels in safeguarding personal information security, regardless of whether fine-tuning-based or zero-shot speech synthesis.

Evaluation on ICL-based TTS models. Many existing TTS models extract timbre representations of target speakers through a speaker encoder, and E2E-VGuard also utilizes this approach by integrating multiple speaker encoders to achieve timbre-level prevention. Additionally, some models obtain information such as timbre features through in-context learning (ICL) rather than using a speaker encoder. Table 3 presents our experimental results on three ICL-based models. The results demonstrate that E2E-VGuard maintains SOTA voice protection performance on ICL-based models and exhibits strong transferability. This effectiveness stems from our speaker encoder ensemble technique, which successfully hides or modifies the timbre information of the original speaker. Consequently, the timbre prevention of the proposed E2E-VGuard does not rely on the specific speaker encoder.

4.4 Evaluation via Commercial APIs

Voice cloning through commercial APIs is relatively convenient, requiring only audio data input to train the target speaker. Once the server generates the trained speaker ID, speech synthesis can be readily implemented. This approach eliminates the need for local model deployment while achieving high-quality synthesis, as the underlying models are black-box systems that demand robust defense mechanisms. We select three common commercial products supporting voice replication, including ByteDance, Alibaba, and MiniMax (represented by company names), for evaluation.

As shown in Figure 2, our experimental results reveal that compared with unprotected audio, E2E-VGuard reduces the average SIM score from 0.689 to 0.203 while increasing WER values. This demonstrates E2E-VGuard's strong transferability, effectively safeguarding voiceprint information even in black-box scenarios involving commercial APIs. These performance improvements across similarity and pronunciation metrics confirm its practical effectiveness for real-world deployment.

4.5 Ablation Study

In the ablation study, we explore the functionality of our proposed optimization objectives and the hyperparameter selection for the trade-off of the perturbative performance and imperceptibility.

Component Analyses. In this part, we explore the role of each component in Eq. (1). We separately investigate the impacts of $\mathcal{L}_{asr}$, $\mathcal{L}_{fea}$, and $\mathcal{L}_{psy}$ & $\mathcal{L}_2$ by removing each component from E2E-VGuard and evaluating the resulting protection effectiveness. Table 4 presents the results of this ablation study. We observe that removing the perceptual optimization module, *i.e.*, $\mathcal{L}_{psy}$ & $\mathcal{L}_2$, achieves better protection but significantly increases perceptual disruption to the original audio due to the lack of perceptual alignment of noise, resulting in a low SNR of 12.942. To directly examine

the effects of $\mathcal{L}_{asr}$ and $\mathcal{L}_{fea}$, we optimize each term individually. When $\mathcal{L}_{fea}$ is removed and only $\mathcal{L}_{asr}$ is retained, the SIM value on the GSV model is relatively high at 0.409, indicating significant leakage of voiceprint information. Conversely, when $\mathcal{L}_{asr}$ is removed and only $\mathcal{L}_{fea}$ is retained, the disruption of text-pronunciation alignment diminishes, with the WER on the GSV model dropping to 49.059%. When all three components are present, they collectively provide effective protection at both the pronunciation and timbre levels while maintaining better imperceptibility of perturbations.

Hyperparameter Selection. In the Eq. (1), we employ hyperparameters α and β to balance the effectiveness of E2E-VGuard's protection and the imperceptibility of the embedded, with values empirically set to 500 and 5×10^{-3} [24], respectively. The value of $\alpha = 500$ is chosen to amplify the loss for speaker identity protection, aligning its optimization scale approximately with that of $\mathcal{L}_{\text{asr}}$. β is used to trade off the protection effectiveness and the perception quality of the perturbation. Specifically, a larger β (such as 5×10^{-2}) yields better perception quality but weaker protection, *e.g.*, the speaker similarity degrading, whereas a smaller β (such as 5×10^{-4}) enhances protection at the cost of reduced perception quality *e.g.*, SNR lower than 15. Through empirical evaluation with various β values, we find that $\beta = 5 \times 10^{-3}$ satisfies an approximate trade-off between protection and quality of perception.

4.6 Robustness Test

In real-world scenarios, strong adversaries can find the obtained dataset with specific modifications, and adversarial techniques may be employed to improve synthesis quality. In this part, following [9, 10], we conduct the robustness test against perturbation removal and advanced data augmentation techniques. Moreover, we evaluate the effectiveness and robustness of E2E-VGuard in the real world.

Perturbation Removal. High-quality speech synthesis often requires high-quality input audio without audible perturbation [10]. Therefore, adversaries may utilize perturbation removal techniques to improve the quality of training samples and weaken unknown strategies users adopt before uploading. We refer to the use of two efficient denoising techniques [10], spectral gating (SG), and a DNN-based model named *denoiser* [44] to denoise each protected audio sample. This experiment is conducted on two models, *i.e.*, VITS and GSV. For the SG method, the WER and SIM on the VITS model are 51.005% and 0.224, respectively, while on the GSV model, the WER and SIM are 31.958% and 0.251, respectively. The *denoiser* can nearly remove the audible background noise. We test the protective performance utilizing the *denoiser* for denoising. On the GSV model, the WER and SIM are 23.10% and 0.243, respectively. The WER and SIM on the GSV model are 34.10% and 0.261, respectively. This shows that even after removing the audible noise, E2E-VGuard can still protect speaker privacy, especially at the level of the speaker identity, with an average SIM value of only 0.238 and 0.252 across these two models using the SG and *denoiser*, respectively. The WER on the VITS model exceeds 50% after denoising using the SG method, effectively disrupting pronunciation.

Data Augmentation. Data augmentation is used to alter the specific structures of embedded perturbations, thereby reducing effects. We consider three categories of data augmentation techniques:

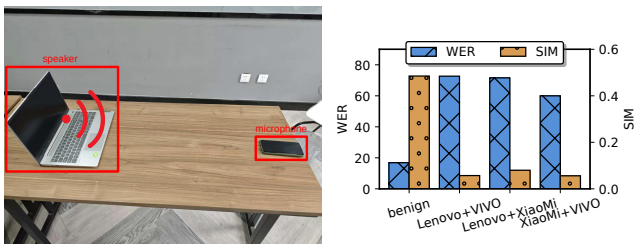
- **Adversarial Defender** [45]: Hussain *et al.* [45] discovered that in the audio field, certain adversarial defense methods can effectively disrupt adversarial audio examples, *e.g.*, proposed E2E-VGuard. These adversarial techniques include: Down-sampling and Up-sampling (RS), Mel-spectrogram Extraction and Inversion (Mel), Quantization-Dequantization (Q-D), and Filtering.
- **Audio Processor** [9]: We consider speech-processing techniques to simulate real-world operations, following Zhang *et al.* [9], including Speed Adjustment (Speed), adding Gaussian noise (Gaussian), Time Masking (TiM), Pitch Shifting (PS), MP3 compression, and Tanh Distortion (Tanh).
- **Filters** [9]: Filtering techniques are commonly used to alter perturbations. We consider three types of filter techniques: Band-Pass Filter (BPF), Low-Pass Filter (LPF), and High-Pass Filter (HPF).

Table 5 shows the results of data augmentations. We observe that the Mel technique significantly improves the intelligibility of synthesized audio, with WER values decreasing by 39.194% and 19.825% on the VITS and GSV models, respectively. However, the SIM values remain high. Although data augmentation can disrupt the perturbation and reduce its protective effect, the embedded perturbation persists, and transforming the audio inherently degrades its quality, *e.g.*, Mel, and Filters.

Real-World Robustness. After users upload audio protected by E2E-VGuard, adversaries may utilize various types of microphones to record and collect the played audio in the real world for voice cloning. To verify the robustness of E2E-VGuard in over-the-air scenarios, we employed different

Table 5: Results under data augmentation and defensive methods. The underline indicates the most significant decreases in protection compared to training without augmentation ("w/ o" in the Table) .

Model	Metric	w/ o	Adversarial Defender [45]				Audio Processor [9]						Filters [9]		
			Resample	Mel	Q-D	Filtering	Speed	Gaussian	TiM	PS	MP3	Tanh	BPF	LPF	HPF
VITS	WER(↑)	96.735	94.827	**55.633**	103.247	93.552	84.101	84.881	95.654	91.796	82.398	95.441	97.258	95.553	136.552
	SIM(↓)	0.113	0.115	0.122	0.082	0.128	0.080	**0.163**	0.098	0.099	0.125	0.143	0.136	0.118	0.045
GSV	WER(↑)	69.148	55.035	**35.210**	75.011	57.497	77.642	44.232	77.165	68.607	41.903	71.446	83.050	52.882	48.036
	SIM(↓)	0.074	0.195	0.158	0.117	0.147	0.044	0.128	0.109	0.049	0.154	0.126	-0.046	0.135	**0.229**



(a) Real-world environment. (b) Results.

Figure 3: Robust test in the real world. (a) shows the experimental environment. (b) The experimental results. "Lenovo+VIVO" represents the "speaker-microphone".

speakers to play the audio as the speaker and different microphones to record the audio as adversaries. We conduct experiments in a quiet environment with background noise averaging 22 dBA (measured by taking the average over 10 seconds). We apply the built-in speaker of a Lenovo Laptop to play the audio and record the audio using VIVO and XiaoMi phones placed approximately one meter away from the speaker to simulate the adversaries as shown in Figure 3a. In each test, we play 10 audio samples and ensure that the speaker's loudness averages 46 dBA. Finally, for the recorded audio, we employ the GSV model for fine-tuning and cloning due to its excellent few-shot cloning capability [2]. The original synthesized results without perturbation after recording are 16.837% and 0.485 of WER and SIM values with high speaker consistency. The average WER and SIM are 72.615% and 0.068, respectively, in Figure 3b, indicating excellent protection of voice timbre and pronunciation in real-world scenarios. Additionally, when we replace the speaker with a mobile device, a XiaoMi phone, and use the VIVO phone as the recording device, the protective effect remains high. This experiment demonstrates the robustness of E2E-VGuard in the real world, as the over-the-air transmission acts as a form of data augmentation [18], and Section 4.6 illustrates the effectiveness of E2E-VGuard in handling data augmentation.

5 Conclusion

This paper focuses on the current mainstream industrial-level and LLM-based TTS models. Considering the more practical scenario of end-to-end speech synthesis, we propose a protection technique, E2E-VGuard, that effectively safeguards audio content from both timbre and pronunciation perspectives. We conduct extensive experiments on various speech synthesis models and multilingual datasets for evaluation. Limitations and future work, *e.g.*, time efficiency and the E2E-VGuard's reliance, are discussed in the appendix.

Acknowledgment

We sincerely appreciate anonymous reviewers for their insightful and valuable feedback. Zhisheng Zhang, Yifan Mi, and Jie Hao are supported in part by the National Natural Science Foundation of China under Grant No. U21B2020 and the Fundamental Research Funds for the Central Universities under Grant No. 2024ZCJH05. Zhisheng Zhang, Jie Gao, and Zhiyong Wu are supported in part by the National Natural Science Foundation of China under Grant No. 62076144 and the Shenzhen Science and Technology Program under Grant No. JCYJ20220818101014030.

References

[1] Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, et al. Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens. *arXiv preprint arXiv:2407.05407*, 2024.

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

[3] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 2019.

[4] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[5] Ailin Huang, Boyong Wu, Bruce Wang, Chao Yan, Chen Hu, Chengli Feng, Fei Tian, Feiyu Shen, Jingbei Li, Mingrui Chen, et al. Step-audio: Unified understanding and generation in intelligent speech interaction. *arXiv preprint arXiv:2502.11946*, 2025.

[6] Jaehyeon Kim, Jungil Kong, and Juhee Son. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *International Conference on Machine Learning*, pages 5530–5540. PMLR, 2021.

[7] Zhiyuan Yu, Shixuan Zhai, and Ning Zhang. Antifake: Using adversarial audio to prevent unauthorized speech synthesis. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 460–474, 2023.

[8] Chien-yu Huang, Yist Y Lin, Hung-yi Lee, and Lin-shan Lee. Defending your voice: Adversarial attack on voice conversion. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 552–559. IEEE, 2021.

[9] Zhisheng Zhang, Qianyi Yang, Derui Wang, Pengyang Huang, Yuxin Cao, Kai Ye, and Jie Hao. Mitigating unauthorized speech synthesis for voice protection. In *Proceedings of the 1st ACM Workshop on Large AI Systems and Models with Privacy and Safety Analysis*, 2024.

[10] Zhisheng Zhang, Derui Wang, Qianyi Yang, Pengyang Huang, Junhan Pu, Yuxin Cao, Kai Ye, Jie Hao, and Yixian Yang. Safespeech: Robust and universal voice protection against malicious speech synthesis. In *34th USENIX Security Symposium (USENIX Security 25)*, Seattle, WA, USA, 2025.

[11] Eberhard Zwicker and Hugo Fastl. *Psychoacoustics: Facts and models*, volume 22. Springer Science & Business Media, 2013.

[12] Yinghao Aaron Li, Cong Han, Vinay Raghavan, Gavin Mischler, and Nima Mesgarani. Styletts 2: Towards human-level text-to-speech through style diffusion and adversarial training with large speech language models. *Advances in Neural Information Processing Systems*, 36, 2024.

[13] Zhisheng Zhang and Pengyang Huang. Hiddenspeaker: Generate imperceptible unlearnable audios for speaker verification system. In *IJCNN*. IEEE, 2024.

[14] Jie Gao, Haiyun Li, Zhisheng Zhang, and Zhiyong Wu. Black-box adversarial defense against voice conversion using latent space perturbation. In *ICASSP*. IEEE, 2025.

[15] Zirui Zhang, Wei Hao, Aroon Sankoh, William Lin, Emanuel Mendiola-Ortiz, Junfeng Yang, and Chengzhi Mao. I can hear you: Selective robust training for deepfake audio detection. In *ICLR*, 2025.

[16] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR, 2023.

[17] Sigurdur Sigurdsson, Kaare Brandt Petersen, and Tue Lehn-Schiøler. Mel frequency cepstral coefficients: An evaluation of robustness of mp3 encoded music. In *ISMIR*, pages 286–289, 2006.

[18] Guangke Chen, Yedi Zhang, Fu Song, Ting Wang, Xiaoning Du, and Yang Liu. A proactive and dual prevention mechanism against illegal song covers empowered by singing voice conversion. In *Symposium on Network and Distributed System Security (NDSS)*, 2025.

[19] Zheng Fang, Tao Wang, Lingchen Zhao, Shenyi Zhang, Bowen Li, Yunjie Ge, Qi Li, Chao Shen, and Qian Wang. Zero-query adversarial attack on black-box automatic speech recognition systems. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 630–644, 2024.

[20] Yunjie Ge, Lingchen Zhao, Qian Wang, Yiheng Duan, and Minxin Du. Advddos: Zero-query adversarial attacks against commercial speech recognition systems. *IEEE Transactions on Information Forensics and Security*, 18:3647–3661, 2023.

[21] Weifei Jin, Yuxin Cao, Junjie Su, Derui Wang, Yedi Zhang, Minhui Xue, Jie Hao, Jin Song Dong, and Yixian Yang. Whispering under the eaves: Protecting user privacy against commercial and llm-powered automatic speech recognition systems. In *USENIX Security*, 2025.

[22] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376, 2006.

[23] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460, 2020.

[24] Jiangyi Deng, Fei Teng, Yanjiao Chen, Xiaofu Chen, Zhaohui Wang, and Wenyuan Xu. {V-Cloak}: Intelligibility-, naturalness-& {Timbre-Preserving}{Real-Time} voice anonymization. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 5181–5198, 2023.

[25] Rui Duan, Zhe Qu, Shangqing Zhao, Leah Ding, Yao Liu, and Zhuo Lu. Perception-aware attack: Creating adversarial music via reverse-engineering human perception. In *Proceedings of the 2022 ACM SIGSAC conference on computer and communications security*, pages 905–919, 2022.

[26] Zhen Ye, Xinfa Zhu, Chi-Min Chan, Xinsheng Wang, Xu Tan, Jiahe Lei, Yi Peng, Haohe Liu, Yizhu Jin, Zheqi DAI, et al. Llasa: Scaling train-time and inference-time compute for llama-based speech synthesis. *arXiv preprint arXiv:2502.04128*, 2025.

[27] Wei Deng, Siyi Zhou, Jingchen Shu, Jinchao Wang, and Lu Wang. Indextts: An industrial-level controllable and efficient zero-shot text-to-speech system. *arXiv preprint arXiv:2502.05512*, 2025.

[28] Hao-Han Guo, Kun Xie, Yi-Chen Wu, and Feng-Long Xie. Fireredtts-1s: An upgraded streamable foundation text-to-speech system. *arXiv preprint arXiv:2503.20499*, 2025.

[29] Xinsheng Wang, Mingqi Jiang, Ziyang Ma, Ziyu Zhang, Songxiang Liu, Linqin Li, Zheng Liang, Qixi Zheng, Rui Wang, Xiaoqin Feng, et al. Spark-tts: An efficient llm-based text-to-speech model with single-stream decoupled speech tokens. *arXiv preprint arXiv:2503.01710*, 2025.

[30] Edresson Casanova, Kelly Davis, Eren Gölge, Görkem Göknar, Iulian Gulea, Logan Hart, Aya Aljafari, Joshua Meyer, Reuben Morais, Samuel Olayemi, and Julian Weber. Xtts: a massively multilingual zero-shot text-to-speech model. In *Interspeech 2024*, pages 4978–4982, 2024.

[31] Shijia Liao, Yuxuan Wang, Tianyu Li, Yifan Cheng, Ruoyi Zhang, Rongzhi Zhou, and Yijin Xing. Fish-speech: Leveraging large language models for advanced multilingual text-to-speech synthesis. *arXiv preprint arXiv:2411.01156*, 2024.

[32] Nari Labs. Dia. `https://github.com/nari-labs/dia`, 2025.

[33] Ziqiang Zhang, Long Zhou, Chengyi Wang, Sanyuan Chen, Yu Wu, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, et al. Speak foreign languages with your own voice: Cross-lingual neural codec language modeling. *arXiv preprint arXiv:2303.03926*, 2023.

[34] Sefik Emre Eskimez, Xiaofei Wang, Manthan Thakker, Canrun Li, Chung-Hsien Tsai, Zhen Xiao, Hemin Yang, Zirun Zhu, Min Tang, Xu Tan, et al. E2 tts: Embarrassingly easy fully non-autoregressive zero-shot tts. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pages 682–689. IEEE, 2024.

[35] Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, Jian Zhao, Kai Yu, and Xie Chen. F5-tts: A fairytaler that fakes fluent and faithful speech with flow matching. *arXiv preprint arXiv:2410.06885*, 2024.

[36] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, et al. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518, 2022.

[37] Hui Wang, Siqi Zheng, Yafeng Chen, Luyao Cheng, and Qian Chen. Cam++: A fast and efficient network for speaker verification using context-aware masking. In *Interspeech 2023*, pages 5301–5305, 2023.

[38] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al. Conformer: Convolution-augmented transformer for speech recognition. In *INTERSPEECH*, 2020.

[39] Somshubra Majumdar, Jagadeesh Balam, Oleksii Hrinchuk, Vitaly Lavrukhin, Vahid Noroozi, and Boris Ginsburg. Citrinet: Closing the gap between non-autoregressive and autoregressive end-to-end models for automatic speech recognition. *arXiv preprint arXiv:2104.01721*, 2021.

[40] Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. Libritts: A corpus derived from librispeech for text-to-speech. *arXiv preprint arXiv:1904.02882*, 2019.

[41] John Kominek and Alan W Black. The cmu arctic speech databases. In *SSW*, pages 223–224, 2004.

[42] Dong Wang and Xuewei Zhang. Thchs-30: A free chinese speech corpus. *arXiv preprint arXiv:1512.01882*, 2015.

[43] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck. Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification. In *INTERSPEECH*, 2020.

[44] Alexandre Defossez, Gabriel Synnaeve, and Yossi Adi. Real time speech enhancement in the waveform domain. In *Interspeech*, 2020.

[45] Shehzeen Hussain, Paarth Neekhara, Shlomo Dubnov, Julian McAuley, and Farinaz Koushanfar. Waveguard: Understanding and mitigating audio adversarial examples. In *30th USENIX security symposium (USENIX Security 21)*, pages 2273–2290, 2021.

[46] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

[47] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in neural information processing systems*, 33:17022–17033, 2020.

[48] Takuhiro Kaneko, Kou Tanaka, Hirokazu Kameoka, and Shogo Seki. istftnet: Fast and lightweight mel-spectrogram vocoder incorporating inverse short-time fourier transform. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6207–6211. IEEE, 2022.

[49] Hubert Siuzdak. Vocos: Closing the gap between time-domain and fourier-based neural vocoders for high-quality audio synthesis. In *Proceedings of the International Conference on Learning Representations*, 2024.

[50] Sang-gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon. Bigvgan: A universal neural vocoder with large-scale training. In *ICLR*, 2023.

[51] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[52] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[53] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The contribution of this paper has been described in the method and experiment sections.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss some limitations originating from our experiments in Appendix A, *e.g.*, subjective biases when conducting human survey.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: This paper contains no theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We detail the design of the method in Section 3, with details including the hyperparameters used, the encoder selected, and the fine-tuning employed described in Section 4.1 and Appendix D to ensure that the experiments can be reproduced.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The datasets we used are open-source, including LibriTTS [40], CMU ARCTIC [41], and THCHS30 [42], and do not contain private data. Our source code link is available at https://github.com/wxzyd123/E2E-VGuard.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The training and test details have been provided in Section 4.1 and Appendix D, *e.g.*, devices, data splits, hyperparameters of training and perturbation generation.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We have considered the possible bias of human experiments in the subjective experiments, so we introduce 95% confidence intervals to enhance the credibility of the conclusions, and the specific results and descriptions are in the Appendix G, Table 10 and Table 11.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We introduce the computing resources of all the experiments in Section 4.1 and the time of speech protection in Appendix A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have considered and followed the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Our experiments do not involve private data, all experiments are run locally, the only calls to commercial APIs we have made in the Appendix A do not affect the company in any way, and all deepfake audio will not be used for any other social activities.

    Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We introduce a protective defensive framework to better protect our personal information, and do not foresee any high risk for misuse of this work.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have cited them appropriately in this paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: This paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: The human perception is a vital perspective of our work, and we introduce the details of participants, including the recruitment and filtering process, the demographic group (*e.g.*, ages), and the compensation (volunteer).

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: This paper has received approvals from the local Human Ethics Research, and detailed ethical considerations about the human study have been discussed in Appendix A.

    Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: Our research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Discussions and Limitations

Ethical Considerations. To verify the subjective perception of E2E-VGuard in the protected and synthesized audio for human ears, we conduct human subjective testing experiments in Appendix G. These experiments have received approval from the local Human Ethics Research Committee. In the questionnaire, all recruited volunteers are anonymous and consented to their answers being used only for academic research. We do not collect any information beyond the content of the questionnaire, and we maintain strict confidentiality regarding the responses. All synthesized audio is uniformly discarded after the completion of the subjective survey to ensure no security risks through leakage. Additionally, the experiments described in Section 4.4 involve testing with commercial APIs. Before using the company's products, we have completed real-name authentication and documented the process. Our experimental testing is conducted internally and will not affect the company's operations or generate unintended usage impacts.

Broader Impacts. This paper primarily focuses on proposing a proactive defense technique. Our intention is positive, aiming to protect individuals' voices from infringement. We will open-source the E2E-VGuard, grant users the right to use it, and sign relevant disclaimer clauses to ensure that their usage behaviors are not related to the designer and publisher of the E2E-VGuard. According to the licensing agreement, this will not affect the normal and positive usage of speech synthesis technology. Additionally, our testing using commercial APIs is conducted solely for local user testing and will not impact any company's products or services. This paper will not result in any negative societal impacts.

Subjective Bias. The conclusions of the subjective experiments are derived from human responses, which can be influenced by the answering circumstances, potentially leading to subjective bias. To reduce the bias in subjective experiments, we calculate the 95% confidence interval of the MOS values following [9] and recruit a sufficient number of volunteers to enhance the reliability of the conclusions. We also implement certain filtering measures to eliminate low-quality responses. According to the conclusions of the subjective experiments, human subjective perception is generally aligned with objective evaluations, indicating that we have minimized the interference of subjective bias on the experimental results.

Limitations of the Target ASR System. In designing E2E-VGuard, we focus on protecting against a specific ASR system, aiming to cause errors in the text recognized by the target ASR. This breaks the alignment between text and pronunciation in the TTS model. The reason we do not pursue a universal approach is that previous methods [19] based on universal attacks typically rely on clustering multiple ASRs, which consumes more time and computing resources. We aim to simplify our system as much as possible to improve the efficiency of the protection. Therefore, we conduct targeted protection using the targeted ASR and then transfer the results to other models. The experiments in Appendix E also demonstrate that our method remains effective when employing different targeted ASR systems.

Time Overhead and Acceleration Strategies. As the defender, we consider scenarios where users upload audio to the internet after E2E-VGuard protects audio samples. We test the average time for E2E-VGuard to protect audio on the LibriTTS dataset using a device equipped with one NVIDIA 4090 GPU with 24 GB of memory. On average, E2E-VGuard takes 97.982 seconds and 111.495 seconds to protect audio in untargeted and targeted settings, respectively. This protection time is on the same level as the baselines, which are 44.871 seconds for AttackVC [8] and 203.248 seconds for AntiFake [7]. The shorter optimization time may allow users to protect the target audio more quickly. The additional time overhead for targeted protection with E2E-VGuard, compared to untargeted protection, mainly comes from selecting a target speaker from the speaker database. Moreover, some acceleration methods can reduce the time overhead, *e.g.*, data batching and multi-GPU parallelization.

Adaptive Adversaries. In the Section 4.6, we evaluate the robustness of E2E-VGuard from three aspects. The experiments demonstrate that E2E-VGuard can resist denoising techniques and remains effective against the adversarial example defense techniques proposed by Hussain *et al.* [45] in the audio domain. Additionally, we test various audio compression and filtering techniques and simulate the adversary's acquisition of audio using different speakers and microphones in real-world scenarios. The E2E-VGuard can still effectively protect the audio. The reason for E2E-VGuard's strong robustness lies in its application of various feature encoders to capture information from the latent space of the audio, enabling perturbations to be better embedded and thus resistant to being disrupted by denoising techniques and others.

Encoder Ensemble. TTS models, especially zero-shot ones, typically design a speaker encoder to extract the timbre embedding of the reference audio. For a specific TTS model, one can perform an adversarial attack on its speaker encoder to mislead the extraction of the target timbre, thereby protecting the original speaker's timbre. In practice, as defenders, we cannot know what type of TTS model the adversary might employ, so our designed proactive defense framework, E2E-VGuard, should possess transferability. Previous research has shown that clustering encoders from different models can achieve outstanding transferability. Based on this, we utilize an encoder ensemble approach and combine it with a feature encoder to better extract and protect the reference speaker's timbre. The encoders we selected are highly representative and can cover mainstream generative architectures and backbones, such as VAE from VITS [6], diffusion model from StyleTTS2 [12], and flow matching from CosyVoice [1].

Eliminating ASR System. In the scenario described in this paper, we have developed an end-to-end fine-tuning method based on an ASR system that does not require manually labeled text. We propose E2E-VGuard for timbre and pronunciation protection. However, assuming the adversary has enough human resources to obtain text through manual labeling, the effectiveness of E2E-VGuard remains a concern. We conduct experiments on the GSV model, using both untargeted and targeted audio protection. For the reference text in fine-tuning, we provide the correct text instead of the text obtained through ASR transcription. The experimental results show that the WER and SIM are 39.659%, 0.161 (T) and 73.784%, 0.278 (UT). This indicates that using clean text can still achieve effective protection at the timbre level, and pronunciation will continue to be affected. This demonstrates that the perturbations added by E2E-VGuard can interfere with the TTS model's learning of pronunciation information. Therefore, E2E-VGuard can still provide some protective effect even when manually labeled correct text is used for fine-tuning.

"Imperceptibility" consideration. In the scenario of our paper, the embedded perturbations should be "harmless" to the original audio, meaning that the original text content remains unaltered and the normal usability of the protected audio is unaffected. "Usability" represents whether the audio can be utilized normally in our daily lives. Rather than requiring perceptual indistinguishability between the protected and original audio. We have verified through both objective (Section 4.2) and subjective experiments (Appendix G) that the perturbations we generated do not cause huge disruptions to the original audio. Moreover, from the robustness perspective, assuming strong adversaries can distinguish embedded perturbations, they can utilize adversarial techniques to improve the performance of the synthesized speech, causing privacy leakage. However, the robustness validated in Section 4.6 ensures that the adversary cannot effectively remove the embedded perturbation, thereby enhancing protection efficacy against speech synthesis. Therefore, even if the adversary perceives the perturbations, the robustness of E2E-VGuard ensures that privacy data is not completely leaked.

B Algorithm

Algorithm 1 provides a detailed illustration of each step that E2E-VGuard utilizes to protect audio. The input data includes the audio to be protected x, a long text Y, the target ASR system, and the optimization numbers max_epoch. The output data is the protected audio x'. Initially, the function `init_perturbation`() is employed to randomly set the initial value of δ, ensuring it stays within $[-\epsilon, \epsilon]$. Subsequently, perturbation optimization is performed for max_epoch steps. In each step, $\mathcal{C}_1$ to $\mathcal{C}_3$ are calculated separately, and their weighted sum yields the objective function value $\mathcal{C}$. For calculating $\mathcal{C}_2$, E2E-VGuard offers two methods, with differing target texts for each case. If untargeted protection is applied, the target text is a segment randomly extracted from the given long text Y, matching the length of the original text $y \leftarrow \mathrm{ASR}(x)$. For targeted protection, the target text corresponds to the transcription of the target audio x_t. Using $\mathcal{C}$, gradient information can be computed to optimize δ, thereby generating the protected audio x'.

Regarding Perturbation Generation. Following the classical Projected Gradient Descent (PGD) [46] algorithm in the adversarial attack domain, we compute the gradient of the loss function for variable x to derive the perturbation: $\delta = -\texttt{sign}(\nabla_x L)$, where $\texttt{sign}(\cdot)$ denotes the sign function and L represents the loss function. Subsequently, δ is projected onto the ϵ-ball constraint, *i.e.*, $\delta = \texttt{Clamp}(-\texttt{sign}(\nabla_x L), -\epsilon, \epsilon)$. Using δ, the protected audio is updated at each step as $x' = \texttt{Clamp}(x + \delta, -1, 1)$, as shown in Algorithm 1.

Algorithm 1: E2E-VGuard.

Inputs: input audio x, text dict Y, ASR system $\mathrm{ASR}(\cdot)$, optimization numbers max_epoch.
Parameters: perturbation boundary ϵ, weight coefficients in Eq. (1) α and β.
Output: protected audio x'.

$\delta \leftarrow \texttt{init\_perturbation}(-\epsilon, \epsilon)$;
$x' \leftarrow x + \delta$;
for $j \leftarrow 1$ **to** max_epoch **do**
  $\mathcal{C}_1 \leftarrow \mathcal{F}\left(\mathrm{ASR}(x'), y_t\right)$;
  if *Untargeted_Sim* **then**
    $y_t \leftarrow Y_{[:|\mathrm{ASR}(x)|]}$;
    $\mathcal{C}_2 \leftarrow \sum_{i=1}^{k} \mathrm{CS}(E_i(x), E_i(x')) + \mathrm{CS}(M(x), M(x'))$;
  else
    $x_t \leftarrow \texttt{select\_target\_speaker}(x)$;
    $y_t \leftarrow \mathrm{ASR}(x_t)$;
    $\mathcal{C}_2 \leftarrow -\sum_{i=1}^{k} \mathrm{CS}(E_i(x_t), E_i(x')) - \mathrm{CS}(M(x_t), M(x'))$;
  end
  $\mathcal{C}_3 \leftarrow \frac{1}{F} \sum_{f=1}^{F} \max\left(0, p_{x'-x}(f) - \theta_x(f)\right) + ||x' - x||_2$;
  $\mathcal{C} \leftarrow \mathcal{C}_1 + \alpha \cdot \mathcal{C}_2 + \beta \cdot \mathcal{C}_3$;
  $\delta \leftarrow \texttt{Clamp}(-\texttt{sign}(\nabla_x \mathcal{C}), -\epsilon, \epsilon)$;
  $x' \leftarrow x + \delta$;
end

Table 6: The detailed comparison of related works and E2E-VGuard.

Method	Target	Type	Waveform	Phrase	Transferability	Imperceptibility	Robustness	Pronunciation
AttackVC [8]	Voice Protection of Identification	AEs	✗	Inference	✗	ℓ_∞ constraint	✗	✗
AntiFake [7]			✓		Encoder Ensemble	Frequency Penalty and SNR	✓	
POP [9]	Voice Protection of Synthesis Quality and Identification	UEs		Fine-tuning	Pivotal Objective	ℓ_∞ constraint		
SafeSpeech [10]						STOI and STFT loss		✓
E2E-VGuard (ours)		AEs		E2E Zero-shot & Fine-tuning	Encoder Ensemble with Feature Extractor	Psychoacoustic Model		

(1) **Waveform**: whether the perturbation is added on original waveform. (2) **Transferability**: the applied approach to enhance perturbation's transferability.
(3) **Robustness**: whether the robustness has been validated.

C Comparison with Related Work

To provide a clearer comparison of the distinctions and advantages between E2E-VGuard and prior works, we present Table 6. This table compares aspects including algorithmic design objectives, types of data protection, whether perturbation is applied on the waveform, targeted speech synthesis types (phrases), transferability, techniques for enhancing imperceptibility, robustness verification, and consideration of pronunciation-level protection.

E2E-VGuard effectively safeguards end-to-end speech synthesis systems, covering both zero-shot and fine-tuning-based scenarios. It integrates an MFCC extractor based on an encoder ensemble to conceal speaker identity at the completed audio feature level. Specifically, E2E-VGuard demonstrates strong adaptability to LLM-based speech synthesis models. Moreover, it employs a psychoacoustic model to minimize human perception of injected noise. In summary, E2E-VGuard achieves a more effective, robust, and perceptually superior audio protection algorithm.

D Details of Experimental Information

In this section, we illustrate the detailed information of selected synthesizers and ASR systems.

Table 7: The detailed information and comparison of selected synthesizers.

	VITS [6]	GSV [2]	CosyVocie [1]	Llasa-1B [26]	Llasa-8B [26]	StyleTTS2 [12]	Index-TTS [27]	FireRedTTS-1S [28]
Type	fine-tuning	zero-shot	zero-shot	zero-shot	zero-shot	zero-shot	zero-shot	zero-shot
Industrial?	✓	✓	✓	✗	✗	✗	✓	✓
LLM?	✗	✓	✓	✓	✓	✗	✓	✓
Backbone	VAE	GPT2 [3]	Transformer	Llama3-1B [4]	Llama3-8B [4]	diffusion model	GPT2 [3]	semantic LM acoustic LM
Vocoder	Hifi-GAN [47]	Hifi-GAN	Hifi-GAN	HifiGAN iSTFTNet [48]	HifiGAN iSTFTNet [48]	Vocos [49]	BigVGAN2 [50]	semantic decoder
RT	2021	2024	2024	2025	2025	2023	2025	2025
Fine-tune	Full (100 / 200)	Full (50 & 25)	Full (20)	LoRA (2)	LoRA (2)	Full (50)	-	-

	Step-Audio-TTS [5]	Spark-TTS [29]	XTTS-v2 [30]	FishSpeech [31]	Dia-1.6B [32]	F5-TTS [35]	E2-TTS [34]	VALLE-X [33]
Type	zero-shot	zero-shot	zero-shot	zero-shot	zero-shot	zero-shot	zero-shot	zero-shot
Industrial?	✓	✗	✓	✓	✗	✗	✓	✓
LLM?	✓	✓	✓	✓	✓	✗	✗	✓
Backbone	Step-Audio [5]	Qwen2.5-0.5B [51]	GPT2 [3]	Llama [52]	Transfomer	DiT [53]	Flow matching Transformer	Codec
Vocoder	HifiGAN	decoder	HifiGAN	Firefly-GAN [31]	DAC decoder	Vocos [49]	BigVGAN [50]	Codec decoder
RT	2025	2025	2024	2024	2025	2024	2024	2023
Fine-tune	-	-	-	-	-	-	-	-

(1) **Type**: whether this model can perform zero-shot TTS. (2) **iSTFT**: inverse Short-Time Fourier Transform. (3) **LLM**: whether LLM component is employed. (4) **Fine-tune**: the fine-tuning type and epochs.

D.1 Details of Synthesizers

To provide a more comprehensive comparison of the models we adopted, we create Table 7, which outlines the following aspects: model type, industrial origin, whether the model is LLM-based, backbone architecture, vocoder used to convert latent variables into perceptible waveforms, release time (RT), and parameter settings employed in the fine-tuning process described in Section 4.2.

In terms of model types, we select VITS, a classic and backbone model requiring fine-tuning, along with other mainstream zero-shot models. Among the models, eight originate from the industry, and most (12 out of 16) are LLM-based. The LLMs utilized include Qwen2.5, Llama 3, Llama, Step-Audio, GPT2, a Transformer-based model, and a Neural Audio Codec. These language models assist the synthesizer in better learning rhythm, prosody, and semantic features. The "Fine-tune" column in the table indicates the implementation details used for validating the end-to-end fine-tuning scenario in Section 4.2: "Full" denotes full-parameter training, while "LoRA" represents using an auxiliary Low-Rank Adaptation (LoRA) adapter to learn input features. Notably, full-parameter fine-tuning of Llasa-8B demands substantial computational resources, whereas LoRA maintains low computational resource requirements while remaining effective. The second row of numbers indicates training epochs, where "100 / 200" in the table represents training 100 iterations for the single-speaker dataset and 200 for the multi-speaker datasets. "50 & 25" means training 50 epochs for GPT and 25 epochs for SoVITS in the GPT-SoVITS model.

D.2 Details of ASR Systems

In real-world scenarios, adversaries may employ different ASR systems to recognize textual information from audio. We briefly introduce the ASR systems considered in the experiments of Section 4.1, and in this section, we present Table 8 to provide detailed comparisons of the ASR systems used in the experiments described in the Appendix E. This includes differences in the acoustic models, loss function types, and recognition performance measured by the WER metric.

From Table 8, we observe that the selected models incorporate two common backbone architectures, *i.e.*, Transformer and CNN, and employ diverse loss function types. We specifically include the Whisper model [16], a multilingual ASR system known for its high recognition accuracy. The largest variant, large-v3, achieves a WER of only 2.7%, making it the best-performing model among those selected. These seven ASR systems across four categories effectively represent current mainstream technologies in the field of automatic speech recognition.

Table 8: The detailed information and comparison of selected ASR systems.

Models	Acoustic Model	Loss Type	WER in test-clean (%)
Wav2vec2 [23]	Transformer & CNN	CTC	3.4
Whisper [16]	Transformer	Cross Entropy	5.0 (base) 3.4 (small) 2.9 (medium) 2.7 (large)
Conformer [38]	Transformer	CTC	3.7
CitriNet [39]	CNN	CTC	4.4

(1) **CNN**: convolutional neural network. (2) **WER in test-clean**: WER value on LibriSpeech test-clean dataset.

Table 9: The protective effectiveness and imperceptibility targeting adaptive ASR models.

Model	Effectiveness				Imperceptibility	
	VITS		GSV			
	WER(↑)	SIM(↓)	WER(↑)	SIM(↓)	SNR(↑)	PESQ(↑)
Whisper-base [16]	99.598	0.144	101.082	0.088	19.362	2.053
Whisper-small [16]	93.996	0.164	103.518	0.138	19.000	2.077
Whisper-medium [16]	126.298	0.171	109.666	0.173	19.236	2.089
Whisper-large-v3 [16]	84.618	0.163	66.534	0.164	19.245	2.019
Conformer [38]	105.145	0.126	81.753	0.180	12.835	1.638
CitriNet [39]	89.520	0.137	59.921	0.302	12.948	1.629

E Adaptive ASR Systems

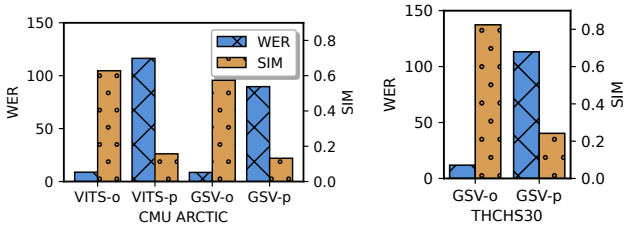
In Section 4.2, Section 4.3, and Section 4.6, we have evaluated the protected effectiveness against the Wav2vec2 model. In the real world, adversaries can employ more types of ASR systems. Therefore, E2E-VGuard should be effective when utilizing different ASR models for text recognition. In this section, we test the protective performance across six other ASR models.

Table 9 shows the protection effectiveness and imperceptibility of E2E-VGuard across different ASR models. The results demonstrate that our method provides strong protection for various ASR models, with WER values consistently above 50%, indicating low intelligibility of synthesized audio, and SIM values below the threshold of 0.25 [43], indicating low similarity to the original audio. Whisper-large-v3, known for its superior text recognition accuracy, ease of utilization, and multilingual capabilities, is widely adopted in the industry. E2E-VGuard also effectively protects against Whisper-large-v3, achieving average WER and SIM values of 75.576% and 0.163, respectively, across two models. Additionally, the imperceptibility of perturbations generated for different models outperforms strong baselines *i.e.*, AntiFake, POP+ESP, and SafeSpeech.

F Multilingual and Multi-Speaker Evaluation

In Section 4.2 and Appendix E, we evaluate the effectiveness and transferability of the proposed E2E-VGuard on a single-speaker English dataset, LibriTTS. However, adversaries may encounter diverse speech samples, including multi-speaker and multilingual scenarios. To address this, we further validate our method on CMU ARCTIC, a multi-speaker dataset, and THCHS30, a multi-speaker Mandarin dataset targeting the Wav2vec2 model. Specifically, we fine-tune VITS and GSV models on CMU ARCTIC while using the GSV model for fine-tuning on THCHS30.

Figure 4a illustrates the experimental results on the multi-speaker dataset ("-o" represents fine-tuning on the original dataset, and "-p" denotes fine-tuning on the protected dataset in the figure). We can find that both models are capable of synthesizing high-quality audio with corresponding speaker timbres on clean samples. After fine-tuning the audio protected by E2E-VGuard, the WER and SIM averaged 103.021 and 0.144, respectively, indicating that E2E-VGuard can effectively prevent the pronunciation and timbre information in a multi-speaker end-to-end fine-tuning scenario. Figure 4b demonstrates the fine-tuning effect of GSV on the THCHS30 dataset, where the recognition ASR system uses a multilingual recognition model, whisper-base, as the target for adversarial attacks.



(a) Multi-speaker dataset. (b) Mandarin dataset.

Figure 4: Test results of Multi-speaker and Mandarin datasets.

Table 10: The subjective evaluation of the ground truth (GT) and E2E-VGuard-protected dataset.

	MOS(↑)
GT	4.788 ± 0.157
E2E-VGuard (UT)	3.522 ± 0.218

Table 11: Human perceptual evaluation of the quality and intelligibility of the synthesized speeches by different training samples.

	MOS(↓)	**Intelligibility**(↓)
clean	4.842 ± 0.123	100.000%
AntiFake	2.055 ± 0.309	99.074%
POP+ESP	$\mathbf{0.851 \pm 0.364}$	89.814%
E2E-VGuard (UT)	2.615 ± 0.331	50.925%
E2E-VGuard (T)	3.060 ± 0.373	**8.333%**

G Human Study

In our previous experiments, we have validated the effectiveness of E2E-VGuard and its perception through objective evaluation metrics. However, we also need to verify the subjective perception of audio by human ears, as synthesized audio in the real world needs to interact with humans. Therefore, this experiment conducts a subjective evaluation to validate human perception and discrimination of synthesized audio, as well as the perception of protected audio.

Recruitment Process. This subjective survey has been approved by the Human Ethics Research Committee at the first author's institution. We create the questionnaire through Credamo and recruit 36 volunteers to participate in the survey, which is comparable to similar studies, such as AntiFake of 24 participants. All volunteers are over 18 years old and possess good English skills. Their average response time is 200.194 seconds.

Filtering. We prohibit volunteers under 18 from participating in the questionnaire. Within the questionnaire, we include two simple random arithmetic questions, and incorrect answers result in rejection. We also filter out all non-serious responses, *e.g.*, the same answers across all questions, or excessively short response times.

Questionnaire Setup. We establish two sections for subjective testing for the synthesized audio and protected audio, with a total of 22 samples.

Task 1: Study on Protected Speech. In the subjective test of protected audio, we select 3 audio samples protected by E2E-VGuard to test naturalness and similarity to the original audio. In order to improve the confidence level of the subjective experiment and reduce the potential bias, we calculate the MOS by taking into account the 95% confidence intervals, which can be found in the previous research [10]. Results in Table 10 show that the MOS of 3.522 ± 0.218 suggests that the embedded perturbations do not significantly reduce normal audio usability, and the distortions are acceptable to human ears when the MOS value surpasses 3.0 [7].

Task 2: Study on Synthetic Speech. In this part, we select 3 synthesized audio samples trained on original samples, baseline-protected methods, and E2E-VGuard-protected samples. Table 11 presents the experimental results for Task 2, revealing that compared to synthesis from the original audio. For synthetic audio evaluation, we utilize audio quality (MOS) and pronunciation intelligibility. Audio quality assesses noise levels and perceptual quality, calculated consistently with Task 1. The ESP method exhibits the worst synthesis quality because its perturbation addition process causes the most severe distortion to the original audio, rendering it unusable. Pronunciation intelligibility

involves presenting participants with both audio and its corresponding prompt text to judge whether the audio content matches the given text. The table results show that E2E-VGuard effectively disrupts the model's original pronunciation patterns: only 50.925% (UT) and 8.333% (T) of participants perceived correct pronunciation alignment with the text, while most participants identified mismatches, demonstrating significant improvement over previous baselines.

Through human study, we observe two key findings: (1) Human auditory perception aligns with objective metrics from prior experiments; (2) Experimental results confirm that E2E-VGuard's noise injection not only bypasses human auditory detection but also substantially reduces the probability of participants being deceived by deepfake audio.