ON THE GENERATION AND REMOVAL OF SPEAKER ADVERSARIAL PERTURBATION FOR VOICE-PRIVACY PROTECTION

Name of author

[1]First Affiliation, CountryX
[2]Second Affiliation, CountryY

ABSTRACT

Neural networks are commonly known to be vulnerable to adversarial attacks mounted through subtle perturbation on the input data. Recent development in voice-privacy protection has shown the positive use cases of the same technique to conceal speaker's voice attribute with additive perturbation signal generated by an adversarial network. This paper examines the reversibility property where an entity generating the adversarial perturbations is authorized to remove them and restore original speech (e.g., the speaker him/herself). A similar technique could also be used by an investigator to deanonymize a voice-protected speech to restore criminals' identities in security and forensic analysis. In this setting, the perturbation generative module is assumed to be known in the removal process. To this end, a joint training of perturbation generation and removal modules is proposed. Experimental results on the LibriSpeech dataset demonstrated that the subtle perturbations added to the original speech can be predicted from the anonymized speech while achieving the goal of privacy protection. By removing these perturbations from the anonymized sample, the original speech can be restored. Audio samples can be found in `https://voiceprivacy.github.io/Perturbation-Generation-Removal/`.

***Index Terms*—** speaker recognition, voice-privacy protection, speaker adversarial perturbation, perturbation removal

1. INTRODUCTION

With the rapid development of *neural network* (NN) in recent years, it has become the default model used in speaker recognition [1, 2] and other applications [3, 4]. In [5], the vulnerability of NNs to adversarial attacks, through subtle perturbation on the input samples, was reported. This seminal work has initiated similar studies in adversarial attacks on speaker recognition. Successful attacks on speaker models have demonstrated their ability to mislead the models into falsely identifying a speaker as someone else [6–9]. These findings further promote the applications of adversarial perturbation in voice-privacy protection [10, 11]. In these studies, the adversarial perturbation technique has shown effectiveness in concealing speaker's voice attributes from unauthorized access with malicious intentions.

Voice attributes play a vital role in speaker recognition applications for security and forensic analysis. The misuse of voice-privacy protection techniques might have adverse consequences. For instance, criminals can exploit these techniques to mitigate forensic analysis and circumvent penalties. In such cases, it is highly desirable to reverse the voice protection and restore the original speaker attributes within the speech. In this context, efforts have been dedicated to the development of *adversarial speaker purification* techniques aiming to neutralize the influence of adversarial perturbations in speaker recognition tasks [12]. Generally, the methods can be categorized as *lossy-preprocessing* [13], *adding noise* [13], *filtering* [13], *denoising* [14] and *generative model* [15]. However, existing works do not aim to fully restore the speech signal. The weakness of existing techniques is manifested in three aspects: 1) existence of residual distortion in the purified speech utterances; 2) degraded performances in automatic speech recognition (ASR), pitch extraction and other downstream tasks. All these purification methods were developed under the premise that the purifiers are unaware of the perturbation generation process. In situations where the restoration of the original speech is intended, these method fail to meet the objective.

In this paper, we examine the reversibility of speaker adversarial perturbations. A well-informed (i.e., white box) setting is investigated, which represents the simplest case. More specifically, we propose a joint training framework in which the removal module is trained simultaneously with the perturbation generator and there well-informed on the perturbation generation process. The removal module estimates the perturbation from the adversarial speech and subtracts the estimated perturbation from the adversarial speech to restore the original speech. In this work, we adopt the symmetric saliency-based encoder-decoder (SSED) proposed in [16] for generating adversarial perturbations. We introduce a joint training strategy where the module responsible for removing perturbations is trained alongside the module used for their generation. The restored speech is evaluated in terms of speech quality and the efficacy of downstream tasks including speaker verification, ASR and pitch extraction. The perturbation pu-

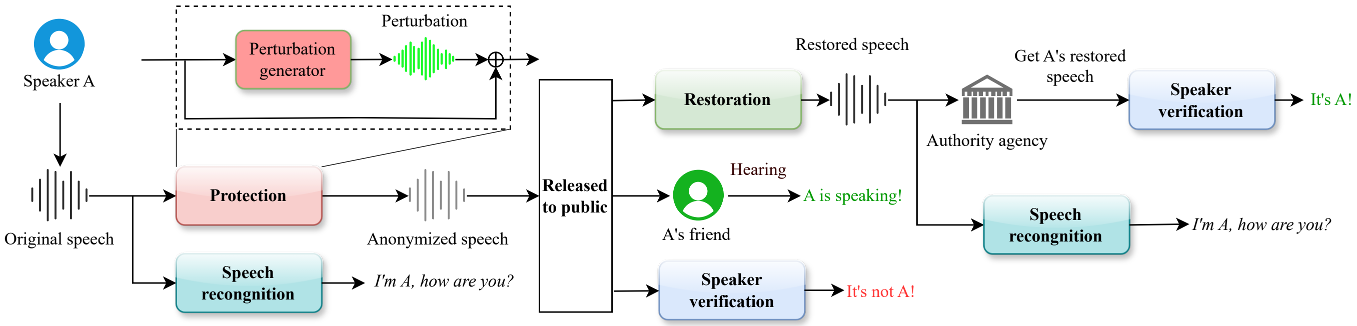

Fig. 1. The process of voice-privacy protection and restoration.

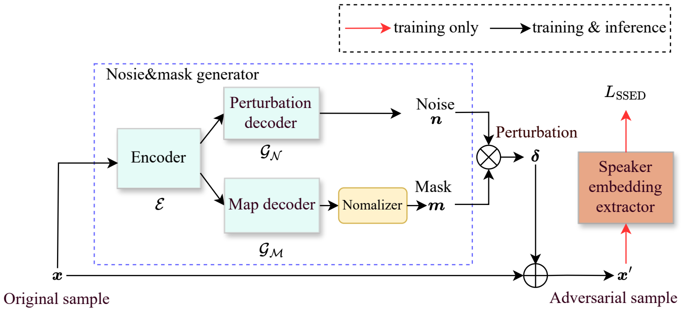

Fig. 2. Framework of symmetric saliency-based encoder-decoder (SSED). The noise and mask are generated by the noise&mask generator, represented in the rectangular box of the blue dotted line. The red lines are applicable only in training, while the black lines are valid in both.

rification methods including the adding noise [13], quantization (lossy-preprocessing [13]), and median smoothing (filtering [13]) methods are examined, providing a reference to our work. The experimental results obtained in speech quality, ASV, ASR, and pitch extraction evaluations validate that the joint training method could effectively restore the original speech.

2. TASK DEFINITION

As shown in Fig. 1, given an original speech of Speaker A, the adversarial perturbation is generated and added to it, resulting in its anonymized version. In this process, the *speaker adversarial perturbation generator* is responsible for generating the perturbation. The anonymized speech can then be released and propagated, for example, through the internet, as exemplified in the figure. The voice protection application can be explained as follows: When using the anonymized utterances, individuals who are familiar with Speaker A, such as A's friend, may still be able to perceive and recognize the speaker's identity by listening. However, when subjected to a speaker recognition algorithm, the identity of Speaker A cannot be recognized.

Meanwhile, consider a scenario: the anonymized speech is used as evidence for courts and public security agencies. It may be necessary to completely remove the influence of the protection to ensure the reliability of the outcomes derived from the speech utterance. In more comprehensive terms, it may be imperative to fully restore the original speaker's information, guaranteeing the confidence of the speaker recognition result obtained from the restored utterance. Furthermore, in regards to the speech content information, the ASR result should align with that acquired from the original utterance. To meet this goal, this paper works on the task that completely restoring the original speech. This paper assesses the restoration capability across three dimensions when compared to the original utterance: 1) speech quality, 2) the speaker attributes as perceived by speaker recognition models, and 3) the speech content and prosody.

3. SSED

In this section, we briefly review the symmetric saliency-based encoder-decoder (SSED) [16] in terms of its architecture and loss function.

3.1. Architecture

The architecture of SSED is shown in Fig 2. It consists of an encoder $\mathcal{E}$, a perturbation decoder $\mathcal{G_N}$, and a saliency map decoder $\mathcal{G_M}$. Given the original speech $\boldsymbol{x}$, it is firstly encoded by the encoder $\mathcal{E}$ into a latent vector $\boldsymbol{y}$. Then $\boldsymbol{y}$ is decoded into a noise vector $\boldsymbol{n}$ by $\mathcal{G_N}$. Parallely, $\mathcal{G_M}$ is applied on $\boldsymbol{y}$, decoding it into the mask representation, which is then normalized to be the mask vector $\boldsymbol{m}$. Finally, $\boldsymbol{\delta}$ is added to $\boldsymbol{x}$, resulting in its adversarial form as follows:

$$\boldsymbol{x}' = \boldsymbol{x} + \underbrace{\epsilon \cdot (\boldsymbol{n} \odot \boldsymbol{m})}_{\boldsymbol{\delta}}, \tag{1}$$

where ϵ denotes the attack intensity.

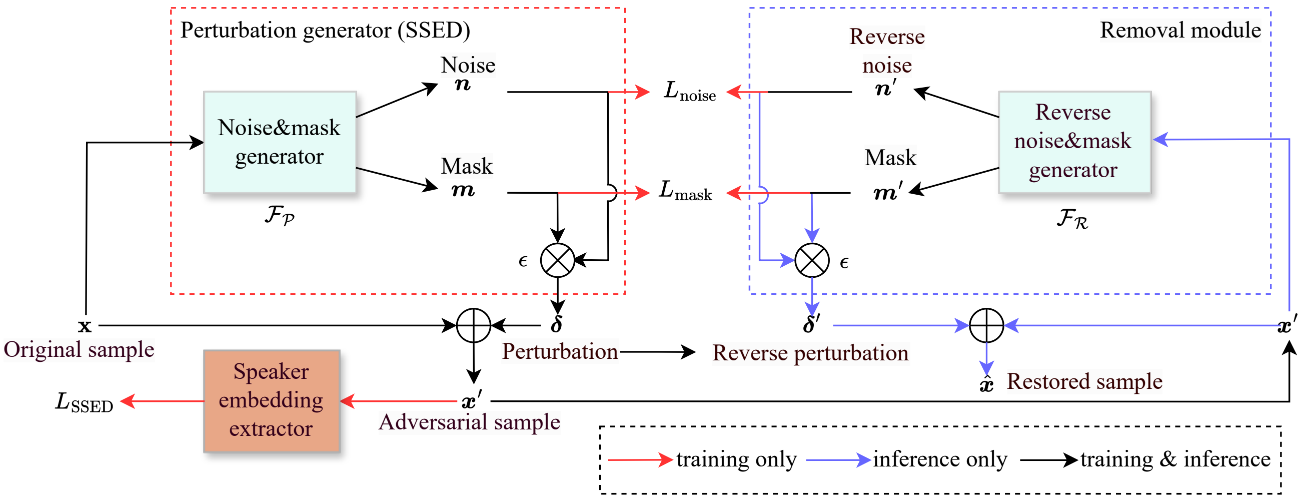

Fig. 3. Model architecture for speaker adversarial perturbation generation and removal. The rectangular boxes of the red and blue dotted lines contain the perturbation generator and the removal module, respectively. The latter takes the adversarial sample $\boldsymbol{x}'$ generated by the former. The noise&mask generator module used in the perturbation generator is inherited from Fig 2. The red and blue lines are applicable only in training and inference, respectively, while the black lines are valid in both.

During the training of the SSED model, the adversarial sample $\boldsymbol{x}'$ is guided by a speaker embedding extractor through the attack mechanism. In inference, given an original speech utterance $\{\boldsymbol{x}_1, ..., \boldsymbol{x}_N\}$ of length N, its adversarial form is obtained as $\{\boldsymbol{x}'_1, ..., \boldsymbol{x}'_N\}$, giving the adversarial utterance.

3.2. Loss function

Our work focuses on protecting the original speaker without the aim of misidentifying it as a target speaker. To achieve this, the angular loss computed on the speaker embedding as defined in [16] is adopted to supervise the adversarial perturbation generation. Given the speaker embedding extractor as presented in Fig 3, the speaker embedding vectors are extracted from the original and adversarial speech utterances, denoted as $\boldsymbol{z}$ and $\boldsymbol{z}'$, respectively. The angular loss is computed as follows:

$$L_{\text{angular}} = \frac{\boldsymbol{z}^{\mathsf{T}}\boldsymbol{z}'}{\|\boldsymbol{z}\|_2\|\boldsymbol{z}'\|_2}, \tag{2}$$

By minimizing L_{angular}, the speaker similarity between adversarial and original sample is reduced, preventing the extraction of the original speaker information by the extractor.

Besides, the loss function to preserve speech quality is defined on the adversarial sample $\boldsymbol{x}'$ and the mask $\boldsymbol{m}$, as follows:

$$L_{\text{quality}} = (1-\alpha)\|\boldsymbol{x}' - \boldsymbol{x}\|_2 + \alpha\|\boldsymbol{m}\|_2. \tag{3}$$

As shown in (3), L_{quality} is composed of two terms. The first one is the Euclidean distance between the adversarial and the original sample. The second term is the L2 norm of the mask vector. Moreover, $\alpha(0 < \alpha < 1)$ is the weight.

Above all, the following loss function is applied in our work for the SSED model training:

$$L_{\text{SSED}} = (1-\beta)L_{\text{angular}} + \beta L_{\text{quality}}. \tag{4}$$

In (4), the trade-off between the adversarial perturbation and the speech quality is achieved by the weight $\beta(0 < \beta < 1)$.

4. PROPOSED METHOD

In this section, given the SSED architecture, we propose a framework whereby the modules responsible for generating and removing perturbations are trained jointly.

4.1. Architecture

The proposed joint-training framework is shown in Fig 3. Given a sample from the original speech $\boldsymbol{x}$, firstly, the noise&mask generator block in SSED is applied as $\mathcal{F}_{\mathcal{P}}$ to generate the noise and mask vectors $\boldsymbol{n}$ and $\boldsymbol{m}$, respectively. Then, the element-wise product between $\boldsymbol{n}$ and $\boldsymbol{m}$ is computed to be the perturbation $\boldsymbol{\delta}$. The adversarial sample $\boldsymbol{x}'$ can be obtained by adding $\boldsymbol{\delta}$ to $\boldsymbol{x}$. Thereafter, a reverse noise&mask gernerator $\mathcal{F}_{\mathcal{R}}$ is proposed which takes the adversarial sample $\boldsymbol{x}'$ as input, and is used to predict the reverse of the noise $\boldsymbol{n}$ as produced by $\mathcal{F}_{\mathcal{P}}$, denoted as $\boldsymbol{n}'$. Meanwhile, the mask vector $\boldsymbol{m}'$ is predicted which is expected to match $\boldsymbol{m}$. The product of $\boldsymbol{n}'$ and $\boldsymbol{m}'$ is computed to be $\boldsymbol{\delta}'$ and is expected to be the reverse of $\boldsymbol{\delta}$. Finally, the *perturbation removal function* is fulfilled by adding $\boldsymbol{\delta}'$ to the adversarial sample $\boldsymbol{x}'$, giving the restored form of the original sample $\hat{\boldsymbol{x}}$. Mathematically, $\hat{\boldsymbol{x}}$ is obtained as:

Table 1. The EERs(%) obtained on the recordings (rec), adversarial (adv) utterances, purified utterances and restored utterances (rst). Adding noise (an) were used purification method. With respect to the enrollment-test trial configurations, five types of tests are included, i.e., *rec-rec*, *rec-adv*, *adv-adv*,*rec-rst*, *rec-an*. The results obtained on the test-clean and dev-clean datasets are presented. Both the white-box and black-box evaluations are included.

Dataset	Gender	White-box (ECAPA-TDNN)					Black-box (ENSKD [17])				
		Protection			Restoration		Protection			Restoration	
		rec-rec	rec-adv	adv-adv	rec-rst	rec-an	rec-rec	rec-adv	adv-adv	rec-rst	rec-an
test-clean	male	1.21	18.62	13.69	1.21	4.82	1.21	12.17	9.56	1.15	3.78
	female	1.64	9.27	18.92	1.64	4.80	2.23	11.78	14.55	2.29	5.57
dev-clean	male	0.50	15.01	13.12	0.50	1.29	0.60	7.62	9.34	0.60	1.29
	female	2.48	23.16	17.09	2.42	7.90	2.59	15.23	13.36	2.77	6.60

$$\hat{\boldsymbol{x}} = \boldsymbol{x}' + \underbrace{\epsilon \cdot (\boldsymbol{n}' \odot \boldsymbol{m}')}_{\boldsymbol{\delta}'}, \tag{5}$$

where ϵ denotes the attack intensity. In our work, the same intensity value is applied for generating both the adversarial perturbation and the reversed perturbation. Besides, in our architecture, $\mathcal{F}_{\mathcal{R}}$ and $\mathcal{F}_{\mathcal{P}}$ share the same structure.

4.2. Loss function

Additionally, guided by the expectation that $\boldsymbol{n}'$ is reverse to $\boldsymbol{n}$, a loss function is defined between them as follows:

$$L_{\text{noise}} = \|\boldsymbol{n} + \boldsymbol{n}'\|_2. \tag{6}$$

Meanwhile, the L2 loss is computed on $\boldsymbol{m}$ and $\boldsymbol{m}'$ as follows:

$$L_{\text{mask}} = \|\boldsymbol{m} - \boldsymbol{m}'\|_2. \tag{7}$$

Combining (6) and (7), the reverse perturbation loss (rpt) is defined as follows:

$$L_{\text{rpt}} = (1-\gamma)L_{\text{mask}} + \gamma L_{\text{noise}}, \tag{8}$$

where the weight $\gamma(0 < \gamma < 1)$ balances the influence of L_{mast} and L_{noise}. Finally, the training loss of the joint-training framework in Fig 3 is defined as follows:

$$L = (1-\theta)L_{\text{SSED}} + \theta L_{\text{rpt}}. \tag{9}$$

In (9), the variable $\theta(0 < \theta < 1)$ is the weight to balance between L_{SSED} and L_{rpt}.

5. EXPERIMENTS

5.1. Datasets

Our experiments were conducted on the LibriSpeech corpus [18]. Specifically, the train-clean-100, train-clean-360, and train-other-500 partitions were used for training. The test-clean and dev-clean datasets were used for evaluation. The recordings were resampled to 16kHz. Following [16], the models worked on waveform samples.

5.2. Speaker embedding extractor

In our experiments, the speaker models adopted the ECAPA-TDNN [2] architecture, which were trained on the Voxceleb1 [19] and 2 [20] datasets, using the open-source toolkit ASV-subtools toolkit[1] [21]. The MUSAN corpus [22] and the RIR datasets [23] were applied for data augmentation.

5.3. Compared methods

Proposed method: In our experiments, the perturbation generator and removal module adopted the same structure as derived from [16]. During training, the encoder of the ECAPA-TDNN model was adopted as the speaker embedding extractor. The weight values in the loss function were: $\alpha = 0.01$, $\beta = 0.007$, $\gamma = 0.8$, $\theta = 0.06$. The learning rate was 10^{-4} and the attack intensity ϵ was 0.05. The model was trained with 30 epochs.

Purification methods: Three speaker adversarial purification methods were examined for reference, including adding noise (SNR $= 25$) [13], quantization (quantized factor $= 2^8$) [13], and median smoothing (kernel size $= 3$) [13], abbreviated as *an*, *qt* and *ms* in the following, respectively. These experiments were performed on adversarial speech generated using the proposed jointly trained perturbation generator. Our purification code refers to the open-source toolkit [2].

5.4. Evaluations

In our ASV evaluations, both the white-box and a black-box evaluations were conducted. In the white-box evaluation, the encoder of the ECAPA-TDNN model was used to extract speaker embedding vectors. In the black-box evaluation, the open-source speaker encoder proposed in [17] was utilized for speaker embedding extraction [3], abbreviated as *ENSKD* in this paper. Cosine distance was adopted as the speaker similarity measurement. Equal error rate (EER) was adopted as the metric. Our ASV evaluations were carried out in a gender-independent manner, with 20 male and 20 female speakers

[1] `https://github.com/Snowdar/asv-subtools`

[2] `https://github.com/speakerguard/speakerguard`

[3] `https://github.com/ductuantruong/enskd`

Table 2. The PESQ, SNR, WERs(%) and pitch correlation (P-Mean for mean, P-Std for standard deviation) results on the recordings (rec) the restored utterances (rst) and the purified utterances. Three purification methods are included: adding noise (an), quantization (qt), and median smoothing (ms). The results are presented on the test-clean dataset.

Evaluation	rec	adv	rst	Purification		
				an	qt	ms
PESQ	4.50	3.67	**4.47**	3.15	3.26	3.31
SNR	∞	32.44	**50.29**	24.15	26.96	15.68
WER	4.08	6.94	**4.08**	5.01	7.15	5.61
P-Mean	1.00	0.98	**1.00**	0.95	0.98	0.95
P-Std	0	0.03	**0.01**	0.11	0.43	0.08

in both the test-clean and dev-clean datasets. 1 target and 30 nontarget trials were composed for each speaker.

Voice-privacy protection: The ability of the adversarial perturbation to protect speaker information was examined with ASV evaluations. With respect to the enrollment-test trial configurations, three conditions were examined, i.e., rec-rec, rec-adv, and adv-adv. Here, *rec* and *adv* are short for recording and adversarial speech, respectively. Table 1 shows the results. Compared to rec-rec, rec-adv and adv-adv achieved higher EER in both white-box and black-box evaluations. This indicates that the perturbation was effective in protecting the speaker attributes within the original recordings.

Speaker information restoration: The assessment of speaker attribute restoration was conducted in the ASV evaluation using the original recordings (rec) for enrollment. The utterances restored using our proposed joint training method (rst) were used as test. Besides, results obtained on the utterances purified by adding noise (an) are presented as they achieved the lowest EERs in our experiments among the examined purification methods. The EERs are shown in Table 1. The EERs in both the rec-an and rec-rst tests are lower compared to rec-adv, showing their efficacy in reducing the influence of the adversarial perturbation on speaker attributes. However, the rec-an evaluations still got higher EERs than rec-rec, inferring that the adding noise purification method was not able to restore the speaker attributes within the speech. Moreover, the utterance restored by the proposed method achieved similar EERs with the recordings in both white- and black-box tests, indicating that the speaker information got restored.

Speech quality restoration: The speech quality was measured with perceptual evaluation of speech quality (PESQ) [24] and SNR. In our evaluations, PESQ adopted the range from -0.5 to 4.5. The values were computed between the original and the test utterances. The results obtained on the adversarial and restored utterances are given in Table 2. The results show that the purification methods yield lower PESQ and SNR values than adversarial speech, suggesting a degradation in speech quality by these methods. The restored speech achieved a PESQ score near the maximum of 4.5 and an SNR value of 50, demonstrating its effectiveness in removing the adversarial perturbation from the speech signal.

ASR evaluation: The Whisper service [4] from OpenAI [25] was called for ASR evaluation. The word error rates (WERs) are given in Table 2. From the table, we can see that the adversarial utterances got higher WERs than the recordings, implying the impact of the adversarial speaker perturbations on speech content. The quantification method raised the WER, while the other two purification methods (adding noise and median smoothing) reduced it, though they did not fully attain the recording's level. However, the restored utterances obtained the same WER as the recordings. This demonstrates the effectiveness of the perturbation removal function in eliminating the influence of perturbations on speech content.

Pitch correlation: Finally, the perturbation removal function in our proposed framework was assessed for its ability to restore prosody information, with pitch serving as the indicator. The statistics (mean and standard deviation) of the pitch correlation were computed as the evaluation metrics. First, for each utterance, the pitch correlation between the recording and the corresponding test utterance was calculated. Then, the mean and standard deviation were calculated across all the utterances in the test set. A higher mean and lower deviation indicate better pitch preservation. The results are included in Table 2, It can be observed that the purification method can damage prosody information. Particularly, the mean values being 1 and deviation values close to 0 obtained by the restored speech demonstrated the effectiveness of the proposed perturbation removal function in restoring the prosody information in speech utterances.

6. CONCLUSIONS&FUTURE WORK

This paper focuses on reversibility of the voice-privacy protection through speaker adversarial perturbation. A well-informed scenario is considered where modules for speaker adversarial generation and removal modules are trained jointly. Our experiments evaluated the restored speech's quality and its effectiveness in downstream tasks, including speaker verification, speech recognition, and pitch estimation. The findings showed that the original speech quality could be restored, and the perturbations' impacts on downstream tasks could be eliminated. Meanwhile, three existing perturbation purification methods were examined where the purifiers were ignorant of the perturbation generation process. Such methods demonstrated an inability to eliminate the perturbation and restore the original speech. Future work will explore the effectiveness on out-of-domain datasets and investigate the capability of denoising methods in perturbation removal.

[4] https://github.com/openai/whisper

7. REFERENCES

[1] David Snyder et al., "Deep neural network embeddings for text-independent speaker verification.," in *Proc. InterSpeech*, 2017, pp. 999–1003.

[2] Brecht Desplanques et al., "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," in *Proc. InterSpeech*, 2020, pp. 3830–3834.

[3] Kaiming He et al., "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[4] Yishuang Ning et al., "A review of deep learning based speech synthesis," *Applied Sciences*, vol. 9, no. 19, pp. 4050, 2019.

[5] Ian J Goodfellow et al., "Explaining and harnessing adversarial examples," *stat*, vol. 1050, pp. 20, 2015.

[6] Xingyu Zhang et al., "Imperceptible black-box waveform-level adversarial attack towards automatic speaker recognition," *Complex & Intelligent Systems*, vol. 9, no. 1, pp. 65–79, 2023.

[7] Abdullah et al., "Hear" no evil", see" kenansville": Efficient and transferable black-box attacks on speech recognition and voice identification systems," in *IEEE Symposium on Security and Privacy (SP)*, 2021, pp. 712–729.

[8] Xu Li et al., "Adversarial attacks on GMM I-Vector based speaker verification systems," in *Proc. ICASSP*, 2020, pp. 6579–6583.

[9] Jiguo Li et al., "Universal adversarial perturbations generative network for speaker recognition," in *Proc. ICME*, 2020, pp. 1–6.

[10] Jingyang Li et al., "Voice guard: protecting voice privacy with strong and imperceptible adversarial perturbation in the time domain," in *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, 2023, pp. 4812–4820.

[11] Shihao Chen et al., "Adversarial speech for voice privacy protection from personalized speech generation," *arXiv preprint arXiv:2401.11857*, 2024.

[12] Haibin Wu et al., "The defender's perspective on automatic speaker verification: An overview," *arXiv preprint arXiv:2305.12804*, 2023.

[13] Guangke Chen et al., "Towards understanding and mitigating audio adversarial examples for speaker recognition," *IEEE Trans. DSC*, 2022.

[14] Yihao Li et al., "A unified speech enhancement approach to mitigate both background noises and adversarial perturbations," *Information Fusion*, vol. 95, pp. 372–383, 2023.

[15] Yibo Bai and Xiao-Lei Zhang, "Diffusion-based adversarial purification for speaker verification," *arXiv preprint arXiv:2310.14270*, 2023.

[16] Jiadi Yao et al., "Symmetric saliency-based adversarial attack to speaker identification," *IEEE Signal Processing Letters*, vol. 30, pp. 1–5, 2023.

[17] Duc-Tuan Truong et al., "Emphasized non-target speaker knowledge in knowledge distillation for automatic speaker verification," in *Proc. ICASSP*, 2024, pp. 10336–10340.

[18] Vassil Panayotov et al., "Librispeech: an asr corpus based on public domain audio books," in *Proc. ICASSP*, 2015, pp. 5206–5210.

[19] A Nagrani et al., "VoxCeleb: a large-scale speaker identification dataset," 2017, pp. 2616–2620.

[20] J Chung et al., "VoxCeleb2: Deep speaker recognition," 2018, pp. 1086–1090.

[21] Fuchuan Tong et al., "ASV-Subtools: Open source toolkit for automatic speaker verification," in *Proc. ICASSP*, 2021, pp. 6184–6188.

[22] David Snyder et al., "MUSAN: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[23] Tom Ko et al., "A study on data augmentation of reverberant speech for robust speech recognition," in *Proc. ICASSP*, 2017, pp. 5220–5224.

[24] Yi Hu et al., "Evaluation of objective quality measures for speech enhancement," *IEEE Trans. ASLP*, vol. 16, no. 1, pp. 229–238, 2007.

[25] Alec Radford et al., "Robust speech recognition via large-scale weak supervision," in *Porc. ICML*, 2023, pp. 28492–28518.