Adversarial Post-Processing of Voice Conversion against Spoofing Detection

Yi-Yang Ding*, Jing-Xuan Zhang*, Li-Juan Liu†, Yuan Jiang*†, Yu Hu*† and Zhen-Hua Ling*
* National Engineering Laboratory for Speech and Language Information Processing,
University of Science and Technology of China, Hefei, P.R.China
E-mail: dingyiy@mail.ustc.edu.cn, nosisi@mail.ustc.edu.cn, zhling@ustc.edu.cn
† iFLYTEK Research, iFLYTEK Co., Ltd., Hefei, P.R.China
E-mail: ljliu@iflytek.com, yuanjiang@iflytek.com, yuhu@iflytek.com

***Abstract*—With the development of speech synthesis and voice conversion techniques, the anti-spoofing task that detects artificial speech signals has received more and more research attentions recently. State-of-the-art spoofing detectors can distinguish the utterances generated by voice conversion from natural ones with high accuracy. This paper proposes a method that improves the ability of voice conversion models against spoofing detection by post-processing the converted speech using a neural network. The network is built using long short-term memories (LSTM) and trained by reducing the distance between the linear frequency cepstrum coefficients (LFCC) of converted utterances and natural references. In our experiments, the SAS dataset was adopted to construct the anti-spoofing system, and the VCTK dataset was used to build voice conversion models. Experimental results show that our proposed method can reduce the detection rate of the anti-spoofing system significantly without losing subjective performance of converted speech.**

***Index Terms*—adversarial examples, voice conversion, anti-spoofing, post-processing, LFCC**

I. Introduction

Voice conversion (VC) is a technique that converts a speaker's voice to another speaker while linguistic information remains. Statistical modeling is an effective approach to convert the acoustic features from the source speaker towards target ones. Since 1990's, the methods based on Gaussian mixture models (GMM) [1, 2] have been proposed. In these methods, the waveforms were reconstructed from converted acoustic features by vocoders, such as Griffin-Lim [3] and STRAIGHT [4]. In recent years, with the development of deep learning, neural networks show strong ability of fitting complex distributions. The voice conversion methods based on deep neural networks (DNN) [5, 6], recurrent neural networks (RNN) [7], and sequence-to-sequence networks [8] have been proposed to describe the mapping relationship between the acoustic features of two speakers and to improve the accuracy of acoustic feature transformation. Meanwhile, neural vocoders, such as WaveNet [9], have also been applied to voice conversion. With the help of these progresses, the quality of converted speech have been improved significantly. For example, the top system in Voice Conversion Challenge 2018 [10] achieved a naturalness mean opinion score (MOS) over 4.0 and a similarity percentage over 80%.

On the other hand, speaker verification [11] is one of the most important biometric authentication means nowadays. With the improvement of speech synthesis and voice conversion techniques, detecting the spoofed audio becomes crucial to speaker verification systems. In previous studies on spoofing detection, various acoustic features have been used. Some methods adopted high time-frequency resolution features, such as linear frequency cepstrum coefficients (LFCC) [12] and constant-Q cepstrum coefficients (CQCC) [13], to build countermeasures. Some methods learned the countermeasures from raw spectral features by deep learning [14]. Then, classifiers are built based on the extracted features. It is conventional to train two GMMs and utilize the log likelihood ratio (LLR) between them for classification. In recent years, convolutional neural networks (CNN) and other deep learning models [15, 16] have been employed and achieved better classification performance than GMMs. With the development of techniques and data resources, the performance of anti-spoofing methods improves rapidly. In ASVspoof2019 [17], the best system obtained an equal error rate (EER) of 0.22% on the logical access task, which means it can effectively distinguish synthetic and converted waveforms from natural ones.

Inspired by recent advances on adversarial example generation [18], we propose a method to improve the ability of voice conversion models against anti-spoofing systems by post-processing the converted speech waveforms using an LSTM-based neural network. In this paper, the baseline countermeasure used by ASVspoof2019 is adopted to build the anti-spoofing system, which utilizes LFCCs as acoustic features together with a GMM classifier. Our previous voice conversion method developed for Voice Conversion Challenge 2018 [19] is employed to build the baseline voice conversion model. In the existing studies on adversarial example generation, both white-box (i.e., the adversary has complete knowledge of the classifier) and black-box (i.e, the adversary has only access to the inputs and outputs of the classifier) settings have been proposed [18, 20]. This paper adopts a semi-white-box setting, which means that the post-processing network is aware that LFCCs are used as classification features

This work was partially funded by the National Nature Science Foundation of China (Grant No. 61871358).

TABLE I
DISTRIBUTION OF THE DATA USED TO BUILD OUR ANTI-SPOOFING SYSTEM.

Subset	Speakers		Utterances	
	Male	Female	Genuine	Spoofed
Training	10	15	3750	12625
Development	15	20	3481	131385
Evaluation	20	26	9200	707817

TABLE II
PERFORMANCE OF DIFFERENT WINDOW LENGTHS IN THE ANTI-SPOOFING SYSTEM.

Window length (ms)	Evaluation set EER(%)	Detection rate of spoofed speech (%)
20	3.06	96.51
50	3.28	96.26

in the spoofing detector, but don't know its model structures and specific parameters. Thus, the post-processing network is trained to reduce the distance between the LFCCs of converted utterances and natural references. Experimental results show that our proposed method can reduce the detection rate of the anti-spoofing system significantly without degrading the subjective performance of converted speech.

The aim of this study is to demonstrate that it is possible to generate adversarial speech against a spoofing detector once the adversary knows what kind of acoustic features are used by the countermeasure. To generalize this adversarial process to the countermeasures with unseen acoustic features and other detection models will be the task of our future work.

This paper is organized as follows. Section II describes the anti-spoofing system adopted in the paper. Section III introduces the details of the proposed method and the experimental results are described in Section IV. Finally, Section V gives the conclusion.

II. ANTI-SPOOFING SYSTEM

An anti-spoofing system was built as the adversarial target of our proposed method. The dataset used to train the anti-spoofing model was the SAS dataset [21] which was an expansion of the ASVspoof2015 dataset. It consisted of both natural speech and spoofed speech. A totally number of 38940 utterances of natural speech were selected from the VCTK dataset [22]. 14 different text-to-speech (TTS) and VC methods were used to produce artificial speech as spoofing attacks. The distribution of the data used to develop the anti-spoofing system is shown in Table I. The training set consisted of 25 people, using two TTS and three VC methods as spoofing attacks. The development set consisted of 35 people, using the same spoofing methods as the training set. The evaluation set consisted of 46 people and all TTS and VC methods were applied as spoofing attacks.

Following the baseline countermeasure in ASVspoof2019, a GMM back-end classifier with LFCC features [17] was adopted for anti-spoofing. For calculating LFCCs, the speech waveforms were first pre-emphasized, and then spectra were calculated with a window length of 20ms and a frame shift of 10ms by 512-point FFT. The order of LFCCs was set to 20. Their dynamic coefficients, i.e. deltas and accelerations, were also calculated. When training the classifier, two GMMs with 512 components were trained using the LFCCs extracted from the natural speech and the spoofed speech in the training set respectively. At the detection stage, the LLRs calculated using these two GMMs were compared with a threshold value for classification. The threshold value was set to obtain the EER point on the development set. The performance of the built anti-spoofing system can be found in the first row of Table II. An EER of 3.06% was obtained on the evaluation set and the detection rate of spoofed speech was 96.51%, which means that 96.51% of the spoofed speech in the evaluation set can be detected successfully.

We also tried to set the window length for calculating LFCCs in the anti-spoofing system as 50ms to make it consistent with the window length of the short-time Fourier transform (STFT) in our proposed post-processing method, which will be introduced in next section. The frame shift was still 10ms and the FFT point was set as 1024. As shown in the second row of Table II, The anti-spoofing system with a window length of 50ms obtained an EER of 3.28% on the evaluation set and the detection rate of spoofed speech was 96.26%. Since its performance was slightly worse on both EER and detection rate than the original configuration of 20ms, we still adopted the anti-spoofing system with 20ms window length in following experiments.

III. PROPOSED METHOD

The overall framework of our proposed method is shown in Fig. 1. Our previous voice conversion method developed for Voice Conversion Challenge 2018 [19] is employed to build the baseline VC system. In this system, 512-dimensional bottleneck features are first extracted from input speech every 40ms by an automatic speech recognition (ASR) model trained using hundreds of hours of speech data with aligned phonetic transcriptions. Then, an LSTM-RNN conversion network is built to predict mel-cepstral coefficients (MCCs) and excitation features from the bottleneck features for each target speaker. Different from previous work [19] which employed a WaveNet vocoder to reconstruct speech waveforms, the conventional STRAIGHT vocoder is adopted in this paper to generate the converted waveforms from the predicted F0 and spectral features in order to improve the efficiency of conducting experiments.

The features used in the anti-spoofing system introduced in Section II are LFCCs, and the spectral features predicted by our voice conversion model are MCCs. Here, the MCC features are calculated from the spectral envelopes analyzed by STRAIGHT, which removes the harmonic structures in STFT spectra by adaptive interpolation. Thus, it is difficult to recover STFT spectra from MCCs and to conduct a differentiable transformation between MCCs and LFCCs. This means that we are not able to set up an adversarial objective function against the spoofing detector to optimize the conversion model

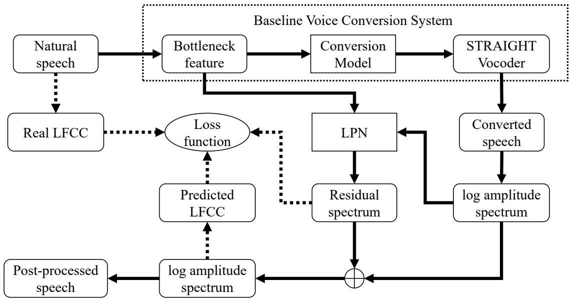


Fig. 1. The framework of our proposed method, where the dotted lines represent the procedures of calculating the loss function for training LFCC-PostNet (LPN) and the solid lines represent the conversion and post-processing procedures.

directly. Therefore, a post-processing method is proposed in this paper which trains an LFCC PostNet (LPN) to post-process the converted waveforms against the spoofing detector.

As shown in Fig. 1, the LPN model adopts the log amplitude spectra calculated from the converted speech by STFT and the bottleneck features used by the conversion model as input. The model structure is composed of an input fully-connected (FC) layer, two LSTM layers, and an output FC layer. The output spectra are used as residuals and added to the input log amplitude spectra to obtain the post-processed log amplitude spectra.

At the training stage, an LPN model is estimated for each target speaker. The bottleneck features extracted from the training utterances of each target speaker are sent into the conversion model to generate speech waveforms. Then, log amplitude spectra are extracted from the converted waveforms by STFT for training the LPN model. The loss function is

$$L = L_{Res} + \alpha * L_{LFCC}. \quad (1)$$

Here, L_{Res} means the mean square error (MSE) between the predicted residual spectra and a zero vector. This term constrains that the post-processing on spectra should be as slight as possible in order to avoid the degradation of naturalness and similarity after post-processing. L_{LFCC} means the MSE between the LFCCs calculated from the post-processed spectra and the natural ones. As introduced in Section I, this paper adopts a semi-white-box setting for adversarial example generation, which assumes that the LPN model knows that LFCCs are used as classification features in the spoofing detector. Thus, L_{LFCC} is expected to improve the ability of converted speech against the anti-spoofing system by reducing the distance between the LFCCs of natural and spoofed speech. α is the weight between these two loss terms which will be investigated in our experiments.

At the conversion stage, the bottleneck features extracted from input source speech are sent into the conversion model and converted waveforms are produced by STRAIGHT vocoder. Then, the bottleneck features and the log amplitude spectra of converted speech are input into LPN. The final speech waveforms are reconstructed from the post-processed amplitude spectra by Griffin-Lim algorithm. In order to make fair comparison between the speech converted by the baseline VC system and the post-processed one, the waveforms produced by the baseline VC system also pass through the Griffin-Lim vocoder in our experiments. A better choice is to employ the same neural vocoder for both the baseline VC system and the post-processed one. This will be a task of our future work.

TABLE III
INFORMATION OF THE SPEAKERS USED IN OUR EXPERIMENTS.

Speaker No.	Age	Gender	Accents	Region
p275	23	M	Scottish	Midlothian
p284	20	M	Scottish	Fife
p283	24	F	Irish	Cork
p288	22	F	Irish	Dublin

TABLE IV
THE ARCHITECTURE OF THE LPN MODEL IN OUR EXPERIMENTS.

Layer	*Unit Number*
FC1	512
LSTM1	512
Projection	256
LSTM2	512
Projection	256
FC2	513

IV. EXPERIMENTS

A. Experimental conditions

As shown in Table III, four speakers in the VCTK dataset [22] were used to build the VC systems in our experiments. Among them, p275 (M1) and p283 (F1) were set as target speakers, and p284 (M2) and p288 (F2) were set as source speakers, which generated four conversion pairs in total. These speakers were also in the evaluation set of the anti-spoofing system, which means they were unseen speakers to the anti-spoofing system. There were about 400 utterances for each speaker, and the waveforms were in 16 kHz sampling rate and 16 bit quantization. 80 percent of the data for each speaker was used as the training set for the voice conversion model and the LPN model, 10 percent of the data was used as the development set, and the data remained was used as the evaluation set. The Librosa toolkit in Python [23] was applied to extract the STFT spectra from natural and converted speech. The frame shift was set to 10ms, and the window length was set to 50ms. The point of FFT was 1024 and the spectrum dimension was 513. The order of LFCCs for training the LPN model was set as 20, and their dynamic coefficients were also calculated.

As mentioned above, the LPN model consisted of two FC layers and two LSTM layers. Its input features were 1025-dimensional vectors composed of 513-dimensional log amplitude spectra and 512-dimensional bottleneck features. As shown in Table IV, the unit number of the input FC layer was 512 and the unit number of each LSTM layer was 512.

TABLE V
MCD (dB), F0 RMSE (Hz) and detection rate (DR) (%) of using different α in LPN training, where VC stands for the results without post-processing.

		VC	0.05	0.1	0.2	0.5
M1	MCD	2.73	2.71	2.70	2.70	2.72
	F0RMSE	22.21	27.74	24.59	27.40	32.22
	DR	86.49	75.68	64.86	43.24	21.62
F1	MCD	2.87	2.85	2.81	2.72	2.67
	F0RMSE	21.42	20.96	18.39	19.94	25.07
	DR	61.90	47.62	42.86	26.19	9.52

Peepholes and projection layer were applied to LSTM and the number of projection units was 256. The unit number of the output FC layer was 513, i.e., the dimension of log amplitude spectra. In our experiments, bottleneck features were normalized to zero mean and unit variance, while spectra and LFCCs were not normalized. Dropout was not applied and the learning rate was set to 0.001 with exponential decay.

B. Effects of weight α in loss function

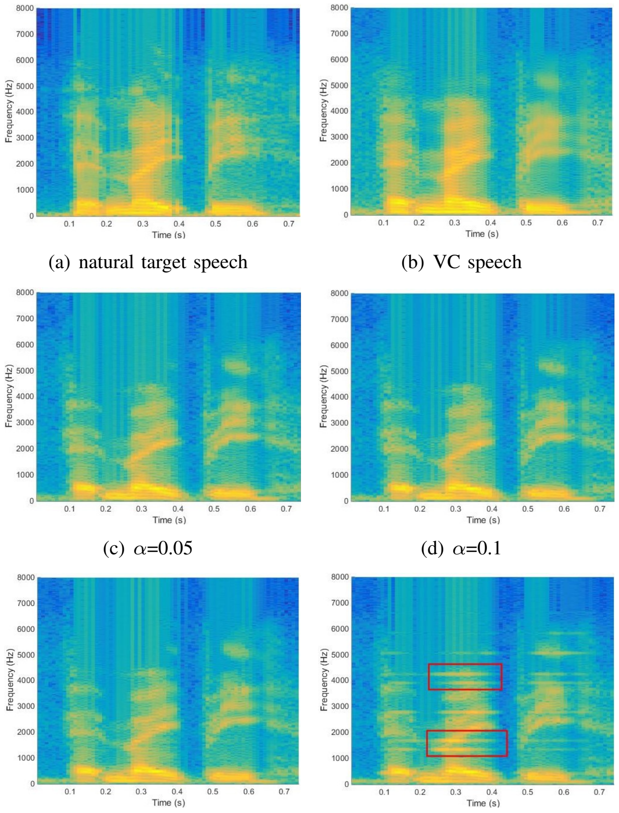
As mentioned in Section III, the weight α is a hyper-parameter in the loss function of LPN and needs to be determined by experiments. In this experiment, four different weights from 0.05 to 0.5 were used to train the LPN models for the two target speakers and their objective performances were compared with the baseline VC models without post-processing. Since VCTK is a non-parallel corpus, in order to calculate the prediction errors of acoustic features, the utterances used for testing in this experiment were generated by the bottleneck features of each target speaker's own evaluation set. Three evaluation metrics were applied. The first one was the detection rate of the converted speech using the anti-spoofing system. Lower rate indicates stronger ability of the VC system against the spoofing detector. The other two metrics were mel-cepstral distortion (MCD) and F0 RMSE of converted speech.

The evaluation results are shown in Table V. From this table, we can see that as the weight increased, the detection rate gradually decreased. This is reasonable since the L_{LFCC} loss in (1) is expected to improve the ability of converted speech against spoofing detectors. As the weight became large, the MCDs of the two target speakers didn't increase, while the accuracy of F0 prediction decreased especially when $\alpha > 0.2$.

The spectrograms of an example utterance generated by different configurations in Table V are illustrated in Fig. 2. From this figure, we can see that the overall formant structures of VC speech were maintained well after post-processing. When $\alpha = 0.5$, some excessive spectral modifications and artificial spectral components can be observed as shown by the red boxes in Fig. 2.(f), which caused the degradation of subjective performance. Therefore, α was finally set as 0.2 in following experiments.

C. Performance of our proposed method

In this experiment, the converted utterances for evaluation were generated using the bottleneck features from the evaluation set of M2 and F2 and the conversion model of M1 and F1. Then, these utterances were post-processed using the LPN models of M1 and F1 respectively[1].



(a) natural target speech

(b) VC speech

(c) α=0.05

(d) α=0.1

(e) α=0.2

(f) α=0.5

Fig. 2. The spectrograms of an example utterance "It may be" generated by different configurations in Table V.

TABLE VI
The detection rates (%) of converted speech before and after post-processing.

Speaker pair	M2_M1	M2_F1	F2_M1	F2_F1
Before	84.62	58.97	94.74	65.79
After	30.77	10.26	68.42	23.68

First, the detection rates of the converted speech before and after post-processing were evaluated and the results are shown in Table VI. From the table, we can see that the detection rates of all four conversion pairs were largely reduced after post-processing, indicating that our proposed method can successfully improve the ability of the VC system against the spoofing detector under a semi-white-box setting.

Then, ABX preference tests were conducted to compare the subjective performance of the converted speech before and after post-processing for the four speaker pairs. In each test, about 30 pairs of test utterances were randomly selected. Each pair of utterances were presented to listeners randomly, who were asked to give their preferences in term of both similarity

[1] Audio demos are available at https://yiyangding.github.io/LFCCPostNet/.

TABLE VII
PREFERENCE TEST RESULTS (%) ON NATURALNESS (NAT.) AND SIMILARITY (SIM.) BETWEEN THE CONVERTED SPEECH BEFORE AND AFTER POST-PROCESSING FOR DIFFERENT SPEAKER PAIRS, WHERE N/P DENOTES "NO PREFERENCE" AND p MEANS THE p-VALUE OF t-TEST BETWEEN TWO SYSTEMS.

		Before	After	N/P	p
M2_M1	Nat.	30.21	28.40	41.39	0.54
	Sim.	30.06	26.59	43.35	0.24
M2_F1	Nat.	19.67	**33.93**	46.40	<0.01
	Sim.	16.54	**30.98**	52.48	<0.01
F2_M1	Nat.	24.11	23.71	52.19	0.87
	Sim.	22.15	23.74	54.11	0.52
F2_F1	Nat.	16.70	**26.57**	56.73	<0.01
	Sim.	16.73	23.20	60.07	0.02

and naturalness. The evaluations were performed on Amazon Mechanical Turk. 20 English native listeners participated in each test and they were asked to use headphones. The results are shown in Table VII. From the table, we can see that for two speaker pairs (M2_M1 and F2_M1), there was no significant preference between the convert speech before and after post-processing in terms of both naturalness and similarity. For the other two speaker pairs (M2_F1 and F2_F1), the converted speech after post-processing got more preference, indicating that the subjective quality of converted speech was improved after post-processing. One possible reason is that LFCC is also a kind of spectral representation and the subjective quality of converted speech may benefit from the reduction of LFCC distortion using the LPN model.

V. CONCLUSIONS

This paper has proposed a method to improve the ability of voice conversion against spoofing detection. An LFCC-PostNet (LPN) is built to post-process the converted speech and its model parameters are estimated to reduce the distortion of LFCC features caused by the conversion model. Experimental results show that our proposed method can effectively improve the ability of converted speech against anti-spoofing systems without decreasing or even slightly improving the subjective quality of converted speech. This paper adopts a semi-white-box setting. To extend our proposed method to black-box settings, such as spoofing detectors with unseen acoustic features and classifiers, is worth further investigation. To experiment with more advanced voice conversion methods and spoofing detection models will also be the tasks of our future work.

REFERENCES

[1] A. Kain, "Spectral voice conversion for text-to-speech synthesis," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* vol. 1, 1998, pp. 285-288.

[2] T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio Speech and Language Processing,* vol. 15, no. 8, pp. 2222-2235, 2007.

[3] D. Griffin and J. Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing,* vol. 32, no. 2, pp. 236-243, 1984.

[4] H. Kawahara, I. Masuda-Katsuse, and A. De Cheveigne, "Restructuring speech representations using a pitch-adaptive timeCfrequency smoothing and an instantaneous-frequency-based f0 extraction: Possible role of a repetitive structure in sounds," *Speech communication,* vol. 27, no. 3-4, pp. 187-207, 1999.

[5] L. Chen, Z. Ling, L. Liu, and L. Dai, "Voice conversion using deep neural networks with layer-wise generative training," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP),* vol. 22, no. 12, pp. 1859-1872, 2014.

[6] H. Zheng, W. Cai, T. Zhou et al., "Text-independent voice conversion using deep neural network based phonetic level features," *2016 23rd International Conference on Pattern Recognition (ICPR),* IEEE, 2016, pp. 2872-2877.

[7] L. Sun, S. Kang, K. Li, and H. Meng, "Voice conversion using deep bidirectional long short-term memory based recurrent neural networks," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* IEEE, 2015, pp. 4869-4873.

[8] J. Zhang, Z. Ling, L. Liu et al., "Sequence-to-Sequence acoustic modeling for voice conversion," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP),* vol. 27, no. 3, pp. 631-644, 2019.

[9] A. V. D. Oord, S. Dieleman, H. Zen et al., "WaveNet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499,* 2016.

[10] J. Lorenzo-Trueba, J. Yamagishi, T. Toda et al., "The voice conversion challenge 2018: Promoting development of parallel and nonparallel methods," *arXiv preprint arXiv:1804.04262,* 2018.

[11] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," *Odyssey,* vol. 14, 2010.

[12] M. Sahidullah, T. Kinnunen, and C. Hanilci, "A comparison of features for synthetic speech detection," *Interspeech,* 2015.

[13] M. Todisco, H. Delgado, and N. Evans, "Constant Q cepstral coefficients: A spoofing countermeasure for automatic speaker verification," *Computer Speech & Language,* vol. 45, pp. 516-535, 2017.

[14] A. Gomez-Alanis, A. M. Peinado, J. A. Gonzalez, and A. M. Gomez, "A light convolutional GRU-RNN deep feature extractor for ASV spoofing detection," *Proc. Interspeech 2019,* pp. 1068-1072, 2019.

[15] G. Lavrentyeva, S. Novoselov, E. Malykh et al., "Audio replay attack detection with deep learning frameworks," *Interspeech,* 2017, pp. 82-86.

[16] Z. Chen, Z. Xie, W. Zhang, and X. Xu, "ResNet and model fusion for automatic spoofing detection," *Interspeech,* 2017, pp. 102-106.

[17] M. Todisco, X. Wang, V. Vestman et al., "ASVspoof 2019: Future horizons in spoofed and fake audio detection," *arXiv preprint arXiv:1904.05441,* 2019.

[18] N. Carlini and D. Wagner, "Audio adversarial examples: Targeted attacks on speech-to-text," *2018 IEEE Security and Privacy Workshops (SPW),* IEEE, 2018, pp. 1-7.

[19] L. Liu, Z. Ling, Y. Jiang et al., "WaveNet vocoder with limited training data for voice conversion," *Interspeech,* 2018, pp.1983-1987.

[20] R. Taori, A. Kamsetty, B. Chu, and N. Vemuri, "Targeted adversarial examples for black box audio systems," *arXiv preprint arXiv:1805.07820,* 2018.

[21] Z. Wu, A. Khodabakhsh, C. Demiroglu et al., "SAS: A speaker verification spoofing database containing diverse attacks," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* IEEE, 2015, pp. 4440-4444.

[22] C. Veaux, J. Yamagishi, K. MacDonald et al., "CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit," *University of Edinburgh. The Centre for Speech Technology Research (CSTR),* 2017.

[23] B. McFee, C. Raffel, D. Liang et al., "Librosa: Audio and music signal analysis in python," *Proceedings of the 14th python in science conference,* vol. 8, 2015.