INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Adversarial Voice Conversion against Neural Spoofing Detectors

Yi-Yang Ding[1], *Li-Juan Liu*[2], *Yu Hu*[1,2], *Zhen-Hua Ling*[1]

[1]National Engineering Laboratory for Speech and Language Information Processing, University of Science and Technology of China, Hefei, P.R.China
[2]iFLYTEK Research, iFLYTEK Co. Ltd., Hefei, P.R.China

dingyiy@mail.ustc.edu.cn, {ljliu, yuhu}@iflytek.com, zhling@ustc.edu.cn

Abstract

The naturalness and similarity of voice conversion have been significantly improved in recent years with the development of deep-learning-based conversion models and neural vocoders. Accordingly, the task of detecting spoofing speech also attracts research attention. In the latest ASVspoof 2019 challenge, the best spoofing detection model can distinguish most artificial utterances from natural ones. Inspired by recent progress of adversarial example generation, this paper proposes an adversarial post-processing network (APN) which generates adversarial examples against a neural-network-based spoofing detector by white-box attack. The APN model post-processes the speech waveforms generated by a baseline voice conversion system. An adversarial loss derived from the spoofing detector together with two regularization losses are applied to optimize the parameters of APN. In our experiments, using the logical access (LA) dataset of ASVspoof 2019, results show that our proposed method can improve the adversarial ability of converted speech against the spoofing detectors based on light convolution neural networks (LCNNs) effectively without degrading its subjective quality.

Index Terms: voice conversion, anti-spoofing, adversarial example generation, white-box, light convolution neural network

1. Introduction

Voice conversion (VC) is a technique that converts a source speaker's voice to a target speaker's voice without changing linguistic information. Since the 1990s, the statistical parameter modeling methods based on Gaussian mixture models (GMMs) [1, 2] and hidden Markov models (HMMs) [3] have been proposed to convert the acoustic features of source speaker toward target speaker. The waveforms were reconstructed from converted acoustic features by vocoders, such as Griffin-Lim [4] and STRAIGHT [5]. With the development of deep learning techniques, neural networks outperform conventional statistical models significantly for describing complex distributions of acoustic features. Deep neural networks (DNNs) [6, 7], recurrent neural networks (RNNs) [8] and sequence-to-sequence networks [9] have been successfully applied to voice conversion. Meanwhile, neural vocoders, such as WaveNet [10], Parallel WaveGAN [11], have also been employed in voice conversion systems. The quality of converted speech has been improved significantly. The top system in Voice Conversion Challenge 2020 [12] achieved a naturalness mean opinion score (MOS) over 4.2 and a similarity percentage over 90%.

On the other hand, the speech generated by voice conversion can be a threat to automatic speaker verification (ASV) systems [13, 14] and detecting such spoofing speech has received more and more research attention nowadays. Some methods adopted high time-frequency resolution features, such as linear frequency cepstrum coefficients (LFCCs) [15] and constant-Q cepstrum coefficients (CQCCs) [16], to build spoofing countermeasures. Some other methods adopted the features encoded from raw spectral features [17]. These extracted detection features are then sent into the classifiers. It is conventional to train two GMMs and utilize the log likelihood ratio (LLR) between them for classification. Convolutional neural networks (CNNs) and other deep learning models [18, 19] have also been employed and achieved better performance. In recent ASVspoof 2019 challenge [20] on evaluating the performance of anti-spoofing systems, the best system obtained an equal error rate (EER) of 0.22% on the logical access (LA) task, which means that most synthetic and converted waveforms can be effectively distinguished from natural ones.

Inspired by recent progress of adversarial example generation [21], this paper makes the first attempt on improving the adversarial ability of voice conversion against neural-network-based spoofing detectors. Similar studies have been conducted against speech classifiers for automatic speech recognition (ASR) [22] and ASV [23]. In our previous work [24], a GMM-based spoofing detector was adopted as the target of adversarial generation, but its structure was simple and its detection performance was inferior to state-of-the-art neural detectors. In this paper, an adversarial post-processing network (APN) is designed to achieve white-box attack against neural spoofing detectors. Its parameters are optimized using an adversarial loss derived from a spoofing detection model together with two regularization losses. Our previous any-to-one voice conversion method for Voice Conversion Challenge 2018 [25] is employed to build the baseline voice conversion system. Two detection models are built based on light convolution neural networks (LCNNs) [26] using two detection features, log amplitude spectra and LFCCs, respectively. Experimental results show that our proposed APN can improve the adversarial ability of converted speech against target spoofing detectors significantly without degrading the subjective quality of converted speech.

This paper is organized as follows. Section 2 introduces our proposed method. Section 3 describes our implementation details and experimental results are shown in Section 4. Finally, Section 5 gives the conclusion.

2. Proposed Method

The framework of our proposed method is shown in Fig. 1. The converted speech generated by a baseline VC system is sent into an adversarial post-processing network (APN) to achieve the white-box attack against a neural-network-based spoofing detection model. The reason that we adopt post-

More details of this study are introduced in a paper that has been submitted to IEEE/ACM Transactions on Audio, Speech, and Language Processing and is currently under review.

http://dx.doi.org/10.21437/Interspeech.2021-948

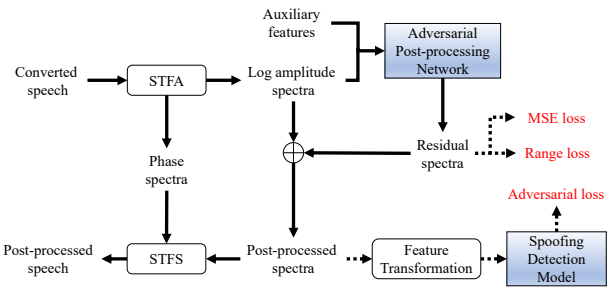


Figure 1: *The framework of our proposed method, where STFA and STFS mean short-time Fourier analysis and short-time Fourier synthesis, the dotted lines represent the procedures of calculating the loss functions for training the adversarial post-processing network (APN) and the solid lines represent the post-processing procedures.*

processing instead of tuning the baseline VC model directly is that there always exist mismatches between the acoustic features employed by VC models and the spoofing detectors.

As shown the Fig. 1, the log amplitude spectra extracted from the converted speech by short-time Fourier analysis (STFA) together with the auxiliary features from the baseline VC system are adopted as the inputs of APN. Residual spectra are predicted by APN and are then added to the input log amplitude spectra to obtain the post-processed spectra, from which the post-processed speech are reconstructed by short-time Fourier synthesis (STFS) combined with the phase spectra of originally converted speech. The Griffin-Lim algorithm [4] is further applied to reduce the mismatch between the post-processed amplitude spectra and the original phase spectra.

When training the APN, the spoofing detection model has been built and its parameters are known since this paper focuses on white-box attack. The detection features used by the spoofing detector are calculated by feature transformation from the post-processed spectra and are then sent into the neural spoofing detector to derive an adversarial loss. The adversarial loss is defined as the negative of the loss function used for training the detection model, which aims to make the post-processed spectra being classified as natural samples by the detection model. Besides, two kinds of regularization losses are designed to guarantee that the spectral modifications are as slight as possible in order to maintain the subjective quality of post-processed speech.

Thus, the overall loss function of training APN is

$$\begin{aligned} L(s, \Delta s) = &- w_a \cdot L_a(F(s + \Delta s)) \\ &+ w_e \cdot L_e(\Delta s) + w_r \cdot L_r(\Delta s), \end{aligned} \tag{1}$$

where s denotes the input log amplitude spectra and Δs stands for the residual spectra predicted by the APN. There are three terms in Eq. (1) with w_a, w_e and w_r as their weights. In the first term, L_a is the loss function for training the detection model based on the golden labels (*spoofing* or *natural*) of training data, and F is the feature transformation mentioned above and depends on the acoustic features used by the spoofing detector. The adversarial loss is the negative of L_a and the golden labels of post-processed spectra are set as *spoofing* when training the APN. Therefore by reducing the adversarial loss, the ability of post-processed speech against the spoofing detector can be improved. The second and third terms are two regularization losses. The second term L_e is the mean square error (MSE) between the predicted residual spectra and a zero vector. The third term L_r is the range of the residual spectra, i.e, the difference between the maximum and the minimum values at each frame, which is designed to avoid that the modifications concentrate on certain narrow frequency bands.

3. Implementation

3.1. Datasets

The data used to build spoofing detection models was the dataset of the logical access (LA) task of ASVspoof 2019 [27]. The bona fide trials in the dataset contained 107 speakers from the VCTK corpus [28]. The spoofed trials were generated by 17 different text-to-speech and voice conversion systems. The utterances were downsampled to 16 kHz with 16-bit quantization. The training set consisted of 2,580 bona fide utterances and 22,800 spoofed utterances generated by 6 attack systems. The speakers of bona fide trials among training, development and evaluation set were not overlapped.

The data used to train the baseline VC system and our proposed APNs was the bona fide utterances from the LA evaluation set of ASVspoof 2019. Among 67 speakers, 8 male and 8 female speakers were chosen as target speakers. There were 95 utterances for each male speaker and 146 utterances for each female speaker. For each target speaker, the ratio between training and development set was 8:2. The source speakers were another 2 male and 2 female speakers, whose utterances were used to form the evaluation set. Therefore, for the built spoofing detectors, the converted utterances were unseen attacks from unseen speakers in our experiments.

3.2. Baseline Voice Conversion System

Our previous voice conversion method [25] was employed to build the baseline voice conversion system. An automatic speech recognition (ASR) model trained with a large speech corpus with phonetic transcriptions was employed as a speaker-independent content feature extractor. The bottleneck features which were treated as content features are extracted from input speech at each frame. Then, a target-speaker-dependent acoustic feature predictor was trained to map the bottleneck features toward acoustic features using the data of the each target speaker. Therefore, an any-to-one voice conversion system can be built without reliance on the training data of source speakers. RNN structure with long short-term memories (LSTM) was adopted to construct the acoustic feature predictor. STRAIGHT was applied to extract the acoustic features including mel-cepstral coefficients (MCCs) and excitation features, and it was also adopted to reconstruct waveforms considering that most training data of spoofing detectors [27] adopted non-neural vocoders.

3.3. Spoofing Detection Models

LCNN [26] has become a popular architecture of neural-network-based spoofing detectors and it was followed to build the spoofing detection models in our implementation. In LCNN-based detectors, max-feature-map (MFM) activation was adopted to reduce the dimensions of channels in convolution layers and to improve the ability of feature selection. Besides, in order to alleviate the overfitting problem, angular margin based softmax (A-softmax) activation [29] was employed instead of conventional softmax. The loss function L_a was the

cross entropy (CE) between the predicted class probabilities and golden labels. In the inference stage, the difference between the logits in the A-softmax layer corresponding to *natural* and *spoofing* classes was used as the score for a test utterance. For natural recordings, the scores should be higher. While for spoofing ones, the scores should be lower.

In our implementation, two types of acoustic features were adopted to build two LCNN-based detection models as follows.

- **FFT-LCNN** Log amplitude spectra were used as detector input. The Librosa toolkit in Python [30] was applied to perform STFA. A Blackman window of 800 points with a frame shift of 160 samples was adopted for short-term analysis by FFT with a frequency points of 1024.
- **LFCC-LCNN** LFCCs were used as input. The filter number was set as 20 and dynamic coefficients were calculated to form 60-dimensional feature. The configurations of FFT were the same as above.

The hyper-parameters of these two models followed the settings in previous work [26]. The length of input segment was set as the first 600 frames of each utterance, while for the utterances shorter than 600 frames, the input was obtained by repeating acoustic features. Adam optimizer was adopted and the learning rate was set as 0.0001 with exponential decay. Normal Kaiming initialization was used and a dropout rate of 0.75 was applied to fully connected (FC) layers to reduce overfitting. In our implementation, the silences at the start and the end of both training and evaluation utterances were trimmed using the FFmpeg tool [31] and the threshold was set as -30 dB.

3.4. Configurations of APNs

The bottleneck features used in the baseline VC system were applied as the auxiliary features in Fig. 1. Similar to the baseline VC system, the APN was also target-speaker-dependent. When training the APN for a specific target speaker, the bottleneck features were extracted from the natural recordings of this target speaker, and the converted utterances were generated using these bottleneck features by the acoustic feature predictor of this speaker to constitute the training data of the APN.

The configurations of STFA for training APNs were the same as the ones in the FFT-LCNN detector. The input of APNs at each frame was 513-dimensional log amplitude spectra, concatenated with 512-dimensional bottleneck features which were normalized to zero mean and unit variance for each dimension. The APN model consisted of two FC layers and two LSTM layers with peepholes and a projection. The unit number of the input FC layer was 512. In each LSTM layer the unit number was 512 and the number of projection units was 256. The output of APN was 513-dimensional residual specctra. Dropout was not applied and an exponential decay was applied to the learning rate.

For each target speaker, two APNs were trained against the two spoofing detection models respectively as follows.

- **FFT-APN** FFT-LCNN was set as the target spoofing detector of white-box attack. The post-processed spectra were sent into FFT-LCNN directly without transformation. The loss weights were empirically set as $w_e = 2$, $w_r = 1$ and $w_a = 1$. The initial learning rate was 0.02.
- **LFCC-APN** LFCC-LCNN was set as the target spoofing detector. LFCCs were calculated from the post-processed spectra by feature transformation. The loss weights were empirically set as $w_e = 20$, $w_r = 1$ and $w_a = 1$. The initial learning rate was 0.002.

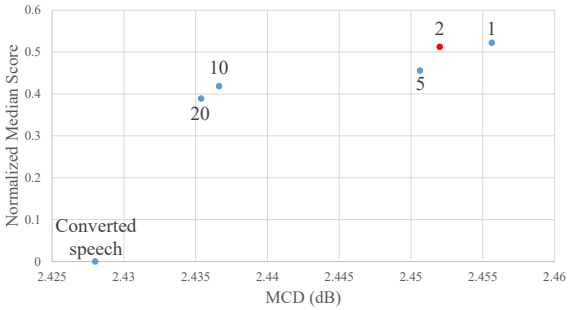


Figure 2: *The MCDs and the normalized median detection scores of post-processed speech on the VC development set when training FFT-APN with MSE loss weight* $w_e \in \{1, 2, 5, 10, 20\}$ *. Here, the range loss weight* w_r *and the adversarial loss weight* w_a *were fixed as 1.*

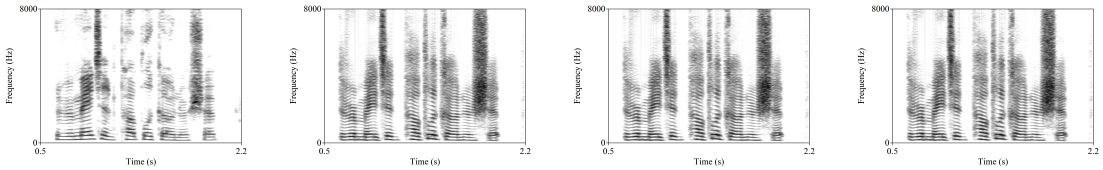
When training both APNs, the loss weights in Eq. (1) were empirically determined by comparing the objective performance of several weight candidates on the development set. The mel-cepstral distortion (MCD) [32] and the normalized median detection score of post-processed speech were adopted as the metrics. The MCCs of post-processed speech and natural recordings were extracted by STRAIGHT for calculating the MCD. Lower MCD implied smaller subjective quality degradation of post-processed speech. Regarding the normalized median detection score, the medians of the scores given by LCNN-based detectors for the originally converted utterances, post-processed ones and natural references of each target speaker were first calculated. Then, for each target speaker, the three median scores were linearly normalized to the interval of $[0, 1]$, where 0 and 1 corresponded to the originally converted speech and the natural speech respectively. Finally, the normalized median score of post-processed speech was further averaged among all target speakers to obtain the evaluation metric. Higher normalized median detection score implied better adversarial ability of post-processed speech. The results of tuning loss weights for training FFT-APN are shown in Fig. 2. It can be seen that as w_e increased, the MCD of post-processed speech was reduced toward that of originally converted speech. As w_e decreased, the normalized median detection score increased which indicated better adversarial ability of post-processed speech against FFT-LCNN. We chose $w_a = 2$ finally as shown by the red dot in Fig. 2.

4. Experimental Results

4.1. VC Performance of Post-processed Speech

Due to the lack of parallel test utterances between source and target speakers, objective acoustic distortions of converted speech can not be calculated. Fig. 3 shows the spectrograms of converting an example utterance in the evaluation set. It can be observed that the spectral modifications made by the two APNs were visually slight.

Furthermore, subjective ABX preference tests were carried out. In order to control the scale of listening tests, 4 target speakers and 2 test utterances were randomly chosen for each of the 4 source speakers. For each APN, an ABX preference test of altogether 32 test utterances were conducted to compare the naturalness and similarity of the converted speech before and



(a) Source (b) Baseline VC (c) FFT-APN (d) LFCC-APN

Figure 3: *The spectrograms of (a) source speech, (b) converted speech by baseline VC and (c)(d) post-processed speech using two APNs for a test utterance.*

Table 1: *Preference test results (%) on naturalness (Nat.) and similarity (Sim.) between the converted speech before and after post-processing using APNs, where N/P denotes "No Preference" and p is the p-value of t-test between two systems.*

		Before	After	N/P	p
FFT-APN	Nat.	33.08	39.66	27.26	0.02
	Sim.	34.91	34.05	31.03	0.75
LFCC-APN	Nat.	33.65	**46.04**	20.31	<0.01
	Sim.	39.17	34.79	26.04	0.12

after post-processing.[1] Each pair of utterances before and after post-processing were presented to listeners in random order, who were asked to give their preferences. The evaluations were performed on Amazon Mechanical Turk and 30 English native listeners participated in each test. The results are shown in Table 1. From this table, it can be seen that there were no significant differences on the naturalness of converted speech for FFT-APN. While the naturalness of converted speech got improved significantly ($p < 0.01$) for LFCC-APN. One possible reason is that some perception-related features were captured via the adversarial learning of LFCC-APN against LFCC-LCNN, which improved the subjective naturalness of converted speech. As for the similarity, there were no significant differences after applying these two APNs. In general, the two APNs in our experiments didn't cause subjective quality degradation.

4.2. Spoofing Performance of Post-processed Speech

An ASV system with the default configurations in ASVspoof 2019 [27] was trained to evaluate the spoofing performance against ASV system of converted speech before and after APN-based post-processing on the evaluation sets of all VC pairs. The labels of spoofing speech were set as *target* in evaluation. The EERs of ASV are shown in Table 2. We can see that the speech post-processed by LFCC-APN slightly sacrificed its ability of spoofing against ASV, while the EER of speech post-processed by FFT-APN was even slightly lower than that of originally converted speech. Anyway, the low EERs in this table indicated that both the converted speech and the post-processed speech were threats to the ASV system.

Further, the adversarial performance of converted speech against spoofing detectors was also evaluated with the metrics of EER and t-DCF [33, 34] on the evaluation sets of all VC pairs. t-DCF combines an anti-spoofing system with an ASV system for evaluation and has a range of $[0, 1]$. Lower EER and t-DCF indicates better performance of spoofing detectors. Since this paper focuses on white-box attack, the detector used for evaluation was the same as the one for training each APN. The results of EERs and t-DCFs before and after post-processing are shown in the Table 3. We can see that FFT-LCNN distinguished baseline VC results from natural recordings accurately with EER of 0.235%, while the performance of LFCC-LCNN was much worse. After applying APNs, the EERs of FFT-LCNN and LFCC-LCNN increased from 0.235% to 10.421% and from 9.514% to 47.181% respectively. The LFCC-LCNN detector almost lost its ability of detecting the spoofing speech after the APN-based white-box adversarial post-processing. Meanwhile, t-DCFs were calculated to evaluate the performance of adversarial post-processing against combined anti-spoofing and ASV. Similar to EERs, t-DCFs also increased significantly after applying APNs as shown in Table 3. In general, the two APNs improved the adversarial ability of converted speech against neural spoofing detectors effectively without sacrificing its subjective performance.

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

Table 2: *The EERs of an ASV system with spoofing utterances before and after post-processing by different APNs.*

	Before	FFT-APN	LFCC-APN
EER(%)	1.324	1.313	1.453

Table 3: *The EERs and t-DCFs before and after post-processing using different APNs. Here, "Detector" means the spoofing detection models used to calculate the evaluation metrics.*

APN	**Detector**	**EER**(%)		**t-DCF**	
		Before	After	Before	After
FFT-APN	FFT-LCNN	0.235	10.421	0.0064	0.2456
LFCC-APN	LFCC-LCNN	9.514	47.181	0.2706	1.0000

5. Conclusion

This paper has proposed a method to generate white-box adversarial examples of voice conversion against neural-network-based speech spoofing detection models. An adversarial post-processing network (APN) is built to deceive an anti-spoofing neural network with an adversarial loss. Experimental results indicated that our proposed APNs can effectively improve the adversarial ability of converted utterances against the speech spoofing detector without degrading their subjective performance. Only white-box attack was studied in this paper. Extending our proposed method to other conditions, e.g., black-box attack, will be a task of our future work. Experimenting with VC systems using neural vocoders, more detection features and more advanced spoofing detection models will also be explored in the future.

6. References

[1] A. Kain, "Spectral voice conversion for text-to-speech synthesis," in *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] T. Masuko, K. Tokuda, T. Kobayashi, and S. Imai, "Voice characteristics conversion for HMM-based speech synthesis system," in *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3. IEEE, 1997, pp. 1611–1614.

[4] 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.

[5] H. Kawahara, I. Masuda-Katsuse, and A. De Cheveigne, "Restructuring speech representations using a pitch-adaptive time–frequency 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.

[6] 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.

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

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

[9] 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.

[10] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," in *9th ISCA Speech Synthesis Workshop*, pp. 125–125.

[11] R. Yamamoto, E. Song, and J.-M. Kim, "Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6199–6203.

[12] Y. Zhao, W.-C. Huang, X. Tian, J. Yamagishi, R. K. Das, T. Kinnunen, Z. Ling, and T. Toda, "Voice conversion challenge 2020: Intra-lingual semi-parallel and cross-lingual voice conversion," *arXiv preprint arXiv:2008.12527*, 2020.

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

[14] N. Dehak, P. J. Kenny, R. Dehak *et al.*, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2010.

[15] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A comparison of features for synthetic speech detection," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[16] 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.

[17] 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.

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

[19] Z. Chen, Z. Xie, W. Zhang, and X. Xu, "ResNet and model fusion for automatic spoofing detection." in *INTERSPEECH*, 2017, pp. 102–106.

[20] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. Evans, T. Kinnunen, and K. A. Lee, "ASVspoof 2019: Future horizons in spoofed and fake audio detection," in *INTERSPEECH 2019-20th Annual Conference of the International Speech Communication Association*, 2019.

[21] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *stat*, vol. 1050, p. 20, 2015.

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

[23] K. Goto and N. Inoue, "Quasi-Newton adversarial attacks on speaker verification systems," in *2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2020, pp. 527–531.

[24] Y.-Y. Ding, J.-X. Zhang, L.-J. Liu, Y. Jiang, Y. Hu, and Z.-H. Ling, "Adversarial post-processing of voice conversion against spoofing detection," in *2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2020, pp. 556–560.

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

[26] G. Lavrentyeva, A. Tseren, M. Volkova, A. Gorlanov, A. Kozlov, and S. Novoselov, "STC antispoofing systems for the ASVspoof2019 challenge," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2019, pp. 1033–1037.

[27] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, N. Evans, M. Sahidullah, V. Vestman, T. Kinnunen, K. A. Lee *et al.*, "ASVspoof 2019: A large-scale public database of synthesized, converted and replayed speech," *Computer Speech & Language*, vol. 64, p. 101114, 2020.

[28] 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.

[29] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 212–220.

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

[31] F. Developers, "ffmpeg tool," 2019.

[32] R. Kubichek, "Mel-cepstral distance measure for objective speech quality assessment," in *Proceedings of IEEE Pacific Rim Conference on Communications Computers and Signal Processing*, vol. 1. IEEE, 1993, pp. 125–128.

[33] T. Kinnunen, K. A. Lee, H. Delgado, N. Evans, M. Todisco, M. Sahidullah, J. Yamagishi, and D. Reynolds, "t-DCF: a detection cost function for the tandem assessment of spoofing countermeasures and automatic speaker verification," in *Speaker Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018.

[34] ASVspoof 2019: the automatic speaker verification spoofing and countermeasures challenge evaluation plan. [Online], Available: http://www.asvspoof.org/asvspoof2019/asvspoof2019_evaluation_plan.pdf.