Interspeech 2024
1-5 September 2024, Kos, Greece

Improving Copy-Synthesis Anti-Spoofing Training Method with Rhythm and Speaker Perturbation

Jingze Lu[1,2], *Yuxiang Zhang*[1], *Zhuo Li*[1], *Zengqiang Shang*[1,*], *Wenchao Wang*[1], *Pengyuan Zhang*[1,2,*]

[1]Key Laboratory of Speech Acoustics and Content Understanding, Institute of Acoustics, Chinese Academy of Sciences, China [2]University of Chinese Academy of Sciences, China

{lujingze, shangzengqiang, zhangpengyuan}@hccl.ioa.ac.cn

Abstract

The rapid development of speech synthesis algorithms poses a challenge in constructing corresponding training datasets for speech anti-spoofing systems in real-world scenarios. The copy-synthesis method offers a simple yet effective solution to this problem. However, the limitation of this method is that it only utilizes the artifacts generated by vocoders, neglecting those from acoustic models. This paper aims to locate the artifacts introduced by the acoustic models of Text-to-Speech (TTS) and Voice Conversion (VC) algorithms, and optimize the copy-synthesis pipeline. The proposed rhythm and speaker perturbation modules successfully boost anti-spoofing models to leverage the artifacts introduced by acoustic models, thereby enhancing their generalization ability when facing various TTS and VC algorithms.

Index Terms: Anti-Spoofing Detection, Copy-Synthesis, Generalization Ability, Logical Access

1. Introduction

Generating speech waveforms with sufficient naturalness is a hot topic in the research field of speech signal processing. With the advent of various carefully designed text-to-speech (TTS) [1] and voice conversion (VC) [2] algorithms, synthesized speech is gradually becoming indistinguishable from natural speech. While providing convenience for our lives, these utterances can also bring some potential security issues [3]. Therefore, constructing robust and reliable spoofing speech detection algorithms is urgently needed.

Thanks to the efforts made by the research community, anti-spoofing countermeasures (CMs) employing various technologies have been successfully constructed. Many effective CMs are designed based on cascade frameworks, leveraging handcrafted features [4, 5, 6] and purpose-designed back-end classifiers [7, 8, 9, 10]. Self-supervised learning (SSL) further enhances the performance of anti-spoofing CMs [11, 12]. Moreover, data augmentation methods [13, 14] and one-class classification methods [15, 16, 17] aim to boost the CMs' generalization ability when facing unseen spoofing attacks, channel coding, compression coding, and other situations.

Although the design of models is crucial for modern DNN-based anti-spoofing CMs, the construction of the training dataset is of equal importance. Typically, anti-spoofing CMs require a training dataset that contains a substantial amount of both natural (bonafide) and synthesized (spoofing) speech. However, the task of detecting spoofing speech has moving targets, given that the technology for synthesizing speech is continually evolving, with numerous new methods emerging annually. Therefore, in real-world applications, constructing the spoofing part of the training dataset proves to be a time-consuming and labor-intensive process [18].

The copy-synthesis approach serves as a simple yet effective solution to the challenge of constructing an appropriate training set [18]. This method employs pre-trained vocoders to convert bonafide utterances to spoofing ones. The effectiveness of the copy-synthesis method can be attributed to it successfully introducing the specific fingerprints [19, 20] carried by vocoders to the generated spoofing speech. An example of such a fingerprint is the aliasing artifact carried by the waveform generators with non-ideal upsampling layers [21], such as HiFi-GAN and VITS. Compared to full-fledged TTS and VC systems, the structure of vocoders is relatively fixed, carrying more consistent artifact information.

Although efficient for building training datasets for the spoofing speech detection (SSD) task, the limitation of the copy-synthesis method is that it only considers the artifacts generated by vocoders. Acoustic models that convert input information into intermediate features may also carry consistent artifacts similar to vocoders. TTS algorithms exhibit deficiencies in predicting rhythm information, while VC algorithms struggle to eliminate the influence of the source speaker. Therefore, we argue that artifacts carried by acoustic models of TTS and VC algorithms are located in rhythm and speaker information, respectively. However, fake utterances generated in a copy-synthesis manner have consistent rhythm and speaker information with natural speech. This paper proposes an optimization method for copy-synthesis based on rhythm and speaker perturbation modules. The proposed modules provide diverse rhythm and speaker information for the generated speech, aiming to guide the classifiers to leverage artifacts of acoustic models. The main contributions of our work include: (1) We analyze the potential locations of artifacts carried by the acoustic models of TTS and VC algorithms. (2) A Rhythm and Speaker Perturbation Copy-Synthesis (RSP-CS) method is proposed in this paper, which achieves performance improvements of CMs in detecting speech generated by various TTS and VC algorithms.

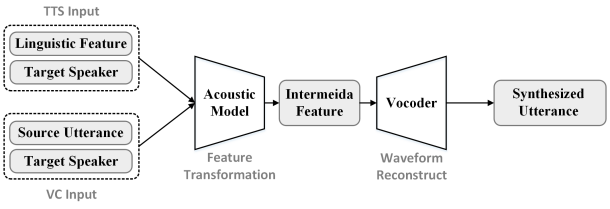

Figure 1: *Basic pipeline of generating synthesized speech.*

* Corresponding author.

10.21437/Interspeech.2024-367

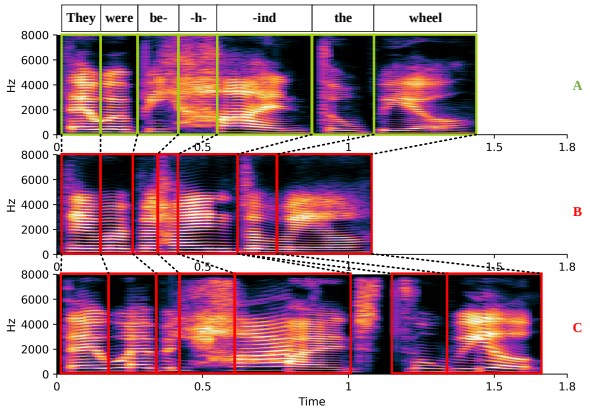

Figure 2: *Spectrograms of three utterances from the ASVspoof 2019 training dataset. A is the spectrogram of a bonafide speech LA_T_4822766, whose textual content is* ***They were behind the wheel****. B and C are the spectrograms of two TTS-generated utterances, with identifiers LA_T_2224464 and LA_T_4184109, respectively. The audio samples B and C share the same speaker and textual content as A.*

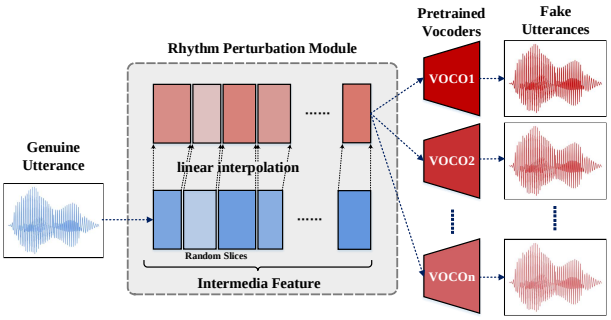

Figure 3: *The pipeline of the copy-synthesis method optimized by the proposed rhythm perturbation module.*

2. Method

This section first analyses the potential artifacts carried by acoustic models of TTS and VC algorithms. Subsequently, an optimization method for copy-synthesis is proposed based on rhythm and speaker perturbation, named the Rhythm and Speaker Perturbation Copy-Synthesis method.

2.1. Aritfacts Carried by Acoustics Models

Figure 1 illustrates the fundamental cascading pipeline of generating spoofing speech, with an acoustic model extracting and transforming the input features into an intermediate feature and a vocoder reconstructing the waveform from these intermediate features. Although the cascading pipelines may not encompass specific end-to-end models, all synthetic algorithms need feature transformation and waveform reconstruction modules.

For constructing more robust anti-spoofing CMs, it is crucial to identify the location of the artifacts. Based on the aforementioned pipeline, the artifacts carried by the synthesized speech can be categorized into those generated by the acoustic models and those produced by the vocoders. The success of the copy-synthesis method could be attributed to its utilizing the vocoder-carried artifacts to detect spoofing speech. However, utilizing the copy-synthesis method to construct the training set also has limitations. The copy-synthesis method ignores artifacts carried by acoustic models. Due to the different inputs, acoustic models of TTS and VC algorithms introduce different kinds of artifacts, which should be discussed separately.

Rhythm is a temporal representation that characterizes how fast the speaker utters each syllable. For TTS algorithms, the inputs do not directly provide rhythm information, which needs to be predicted by the acoustic model. However, the rhythm of speech could be impacted by multiple factors. It is challenging for the generation model to perfectly predict the rhythm through limited input information. Figure 2 shows the spectrum of three utterances from the ASVspoof 2019 dataset. One of these utterances is bonafide, while TTS algorithms generate the other two. Despite sharing the same speaker and textual content, these utterances exhibit significant differences in rhythm information. Therefore, compared to natural speech, TTS-generated waveforms tend to carry rhythm artifacts.

For the VC algorithms, rhythm artifacts are not introduced since the source speech provides the rhythm information. However, the acoustic models of VC algorithms need to convert the source speaker to a new one. Such a process struggles to eliminate the source speaker's impact completely. Based on the above analysis, we argue that the acoustic model of TTS introduces rhythm artifacts, while VC's acoustic model brings speaker artifacts.

2.2. Rhythm Perturbation Module

In the acoustic models of TTS algorithms, the duration predictor module is of significant importance as it predicts the duration of each phoneme. However, due to factors such as overfitting of the training data, the results given by the duration prediction module often deviate from the speech produced by humans. This is the reason for the occurrence of rhythm artifacts.

In section 2.1, we categorize the artifacts introduced by acoustic models into rhythm artifacts and speaker artifacts. The synthesized speech generated in the copy-synthesis manner has a consistent rhythm with natural speech, making it challenging for anti-spoofing CMs to leverage rhythm artifacts. This paper aims to propose an optimized copy-synthesis method to direct the attention of anti-spoofing CMs toward the rhythm artifacts produced by the acoustic models of TTS algorithms. However, if only a few limited TTS algorithms are used to generate phoneme durations when simulating rhythm artifacts, it can also lead to overfitting. Therefore, our goal is to guide the CMs in learning the rhythm features of bonafide speech. Inspired by [22], we introduce a rhythm perturbation module (RPM) to enhance the copy-synthesis pipeline. The proposed module is a random sampling module.

Figure 3 shows the total pipeline of the proposed RPM-based method. In the copy-synthesis pipeline, the intermedia features of genuine speech are first extracted. Next, these features are sent to an RPM module to add rhythm artifacts. The proposed RPM module divides intermeida features into segments, whose length is randomly uniformly drawn from 19 to 32 frames. Each segment is resampled using linear interpolation with a resampling factor randomly drawn from 0.5 (compression by half) to 1.5 (stretch) [22]. Finally, the segments are combined and fed into a pre-trained vocoder to generate synthetic speech. Figure 4 illustrates the fake utterances generated with and without the rhythm perturbation module, where significant differences exist between their rhythms.

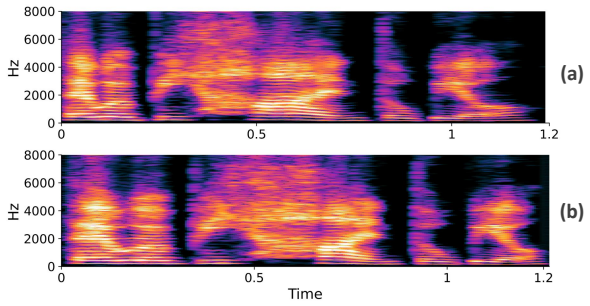

Figure 4: *Spectrums of copy-synthesis fake speech generated with (a) and without (b) the proposed RPM.*

2.3. Speaker Perturbation Module

VC algorithms typically contain an analysis module for extracting intermediate features from the source speech and a mapping module that projects the source speaker onto the target speaker. The artifacts introduced by VC are primarily located on the speaker information. The reason is that VC algorithms need to eliminate the speaker information of the source speech and add the target speaker feature. Such a process may result in some residual of the source speaker [23].

Like the aforementioned RPM, we propose a speaker perturbation module (SPM) to guide the CMs in learning the rhythm features of bonafide speech. To avoid the influence of specific VC algorithms, the proposed SPM is based on the manipulation of the McAdams coefficient (MC) [24]. The MC-based method is a signal processing technique widely used in the speaker anonymization (SA) task, which aims to suppress specific information about the source speaker. Requiring no training data, the MC-SA algorithm can manipulate the positions of formant frequencies of speech signals at the frame level. The proposed SPM adopts the same pipeline of MC-SA algorithm as in [25], with the McAdams coefficient α of each utterance randomly selected from a range of the uniform distribution, i.e., $\alpha \in \mathrm{U}(\alpha_{min}, \alpha_{max})$. In the process of generating fake utterances utilizing SPM, the waveform output by SPM is extracted to intermedia features. Subsequently, the intermediate features are fed into the pre-trained vocoders for waveform reconstruction, thereby introducing vocoder artifacts.

3. Experiments Setup

3.1. Datasets and Metrics

The proposed RSP-CS method is a negative sample generation method, which synthesizes spoofing samples to train CMs, utilizing bonafide speech. To ensure a fair comparison, we follow the copy-synthesis method from [26] and use the same pre-trained vocoders[1] to generate spoofing samples, all of which take mel-spectrum as input. Bonafide utterances from the ASVspoof 2019 Logical Access (19LA) [27] are used to construct the baseline copy-synthesis training dataset and our proposed RSP-CS training dataset, without introducing out-of-distribution (OOD) bonafide samples. The 19LA dataset is influential in spoofing speech detection, with many classic CMs trained on it. Bonafide utterances from 19LA are collected from the Voice Cloning Toolkit (VCTK) corpus [28], which is a multi-speaker English speech database recorded in a hemianechoic chamber.

For evaluation, to measure the generalization ability of anti-spoofing CMs fairly, we conduct experiments on several different datasets. One of them is the eval set of 19LA. Fake utterances in the 19LA eval set are generated by 13 different TTS and VC systems, where A07-A16 are TTS attack models, and A17-A19 are VC attack models. The ASVspoof 2021 Logical Access (21LA) dataset [29] uses the same attack strategies as the 19LA dataset, while its utterances are transmitted across real telephony systems. The ASVspoof 2021 Deepfake (21DF) [29] dataset collects about 600K utterances processed with various lossy codecs typically used for media storage, which is an influential dataset for generalization validation. 21LA and 21DF have hidden tracks in which the non-speech segments are trimmed. In addition, the proposed method is validated on the InTheWild dataset [30], which is a more challenging dataset whose data is collected from the real world. Table 1 presents statistics of datasets.

In this work, the evaluation metric used is equal error rate (EER) $\downarrow$, which is widely used in challenges and research for the task of spoofing speech detection. EER is defined as the point where the false acceptance rate (FAR) and the false rejection rate (FRR) are equal.

Table 1: *Statistics of the evaluation datasets.*

Dateset	Bonafide	Spoof	Total
19LA	7,355	63,882	71,237
21LA	14,816	133,360	148,176
21DF	14,869	519,059	533,928
IntheWild	11,816	19,963	31,779

3.2. Details of Systems Implementation

A pre-trained Wav2Vec 2.0 model [31] is chosen as the front-end, and AASIST structure [10] is chosen as the back-end. The pre-trained Wav2Vec 2.0 model is optimized jointly with the AASIST back-end during the training process. The detailed description of the model architecture can be found in [12]. All models are trained with Adam optimizer [32] with $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 10^{-8}$ and weigth decay 10^{-4}. Cross Entropy loss is adopted as the loss function. The learning rate is fixed at 10^{-6}. The training process is conducted over 100 epochs and a batch size of 32. Rawboost [13] is used as the data augmentation method during the training process.

4. Result and Analysis

4.1. Main Results

To evaluate the proposed RSP-CS method, we first conduct experiments on the evaluation sets of 21LA, which have known attacking algorithms, thus enabling a straightforward performance comparison on detecting TTS-generated and VC-generated utterances. Utterances from the 21LA dataset are transmitted over different codec algorithms, weakening vocoder artifacts in the spectrum. The baseline models are trained using Voc.v3 and Voc.v4 [26] from [26], which are training datasets generated by the copy-synthesis method. Voc.v3 and Voc.v4 consist of bonafide speech from 19LA and spoofing speech generated through four different pre-trained vocoders. Voc.v3 and

[1] https://github.com/nii-yamagishilab/project-NN-Pytorch-scripts/tree/master/project/09-asvspoof-vocoded-trn

Table 2: *Performance of anti-spoofing CMs trained with fake utterances generated by the proposed RP-CS, SP-CS, and RSP-CS methods. Voc.v3 and Voc.v4 are training sets generated with four different pre-trained vocoders [26]. EER(%) ↓ is used as the metric.*

training set	TTS	VC	Total	TTS attacks										VC attacks		
				A07	A08	A09	A10	A11	A12	A13	A14	A15	A16	A17	A18	A19
Voc.v3 [26]	15.00	7.82	13.61	9.48	8.49	4.06	31.11	11.70	14.89	6.01	14.11	12.37	17.99	4.89	4.17	10.82
Voc.v3+RP-CS	**10.04**	8.63	**9.69**	5.41	2.46	1.44	24.84	5.69	11.18	3.20	7.23	8.77	7.05	3.18	2.64	14.49
Voc.v3+SP-CS	14.07	**6.23**	13.09	11.63	10.18	7.29	23.14	13.91	13.02	10.22	13.94	13.29	13.04	4.22	4.29	8.93
Voc.v3+RSP-CS	12.38	7.51	11.04	6.75	5.45	2.68	29.21	11.67	11.35	4.41	10.12	9.71	13.11	3.86	3.23	10.61
Voc.v4 [26]	13.90	7.07	12.57	8.81	5.52	3.64	27.93	11.73	13.41	3.72	13.39	9.62	16.81	3.24	3.57	9.80
Voc.v4+RP-CS	**10.43**	7.15	**9.70**	7.33	3.50	3.04	25.38	7.44	11.47	4.99	6.72	6.72	8.87	2.92	3.29	10.89
Voc.v4+SP-CS	14.12	**6.05**	12.88	10.47	7.21	4.95	27.62	12.54	14.16	6.37	13.94	11.18	15.78	2.57	3.00	8.76
Voc.v4+RSP-CS	10.65	6.67	10.01	7.03	3.96	3.14	24.93	9.01	10.12	2.72	9.85	8.62	10.73	2.51	2.46	10.58

Voc.v4 differ in the datasets utilized for training the vocoders.

For comparison, we train spoofing speech detection models using the training dataset generated by the proposed RSP-CS method. The same pre-trained vocoders as Voc.v3 and Voc.v4 are utilized when generating the fake utterances. Utilizing the RSP-CS method, we create three training datasets to conduct ablation experiments on the proposed rhythm and speaker perturbation modules. In the RP-CS dataset, half of the data employs the RPM, whereas the remaining half solely passes through the pre-trained vocoders. Similar to the RP-CS dataset, only half of the fake utterances in the SP-CS dataset utilize the SPM. In the RSP-CS dataset, both perturbation modules are utilized, and each of the three types of fake utterances, generated with SPM, with RPM, or without both, constitutes one-third of the total. To ensure the fairness of the experimental results, the total number of spoofing speech in the training sets is kept constant.

Results are illustrated in Table 2. For the 21LA dataset, CMs trained with the RP-CS training dataset obtained significant performance improvement, which is in relation to the successful detection of TTS-generated speech. For all TTS-generated speech, CM trained on the RP-CS Voc.v3 training dataset achieves an absolute improvement of 4.96% in EER compared to the baseline. The proposed RPM method positively affects the detection of all ten types of TTS attacks A07-A16. RPM-enhanced fake data enables the detection system to focus more on the rhythm artifacts, thereby improving performance under complex channels. Meanwhile, for VC-generated speech, the performance of RP-CS is slightly decreased. As discussed in this paper, VC-generated utterances do not exhibit rhythm artifacts. In contrast, CM trained on the SP-CS training dataset demonstrates excellent performance in detecting VC-generated utterances while not showing enhancement in the detection of TTS-generated utterances. This aligns with the prior analysis that there are disparities in the speaker information between VC-generated speech and bonafide speech. Compared to the baseline, introducing RPM and SPM at the same time can enhance the CM's performance for both TTS and VC attacks. The experimental results show a consistent trend for the two sets of pre-trained vocoders, Voc.v3 and Voc.v4.

4.2. Generalization Ability

Tabel 4.2 shows the comparison of the baseline method and the proposed RSP-CS method on multiple datasets. For all four evaluation datasets, the proposed achieves significant performance improvement. The 21DF and InTheWild datasets are commonly used to evaluate the generalization ability of CMs. The 21DF dataset contains more than 100 different spoofing attack algorithms, and the InTheWild dataset contains bonafide and spoofing speech collected from the real world. On these two evaluation sets, the RSP-CS method achieves an absolute performance boost of 1.00% and 1.62%, respectively. Experimental results demonstrate that our proposed method successfully enhances the detection performance of the anti-spoofing CMs against various TTS and VC attacks. However, employing just one of either RPM or SPM cannot ensure a performance enhancement on these datasets. Such instability is associated with the proportions of TTS and VC attacks in the evaluation datasets. An example is that most of the fake utterances in the 21DF dataset are generated by VC algorithms. Therefore, SP-CS outperforms RP-CS on 21DF.

Table 3: *EERs (%) ↓ of baseline method and the proposed perturbation method on different datasets.*

Eval Set	Baseline	RP-CS	SP-CS	RSP-CS
19LA	3.12	3.20	**1.07**	2.54
21LA	13.61	**9.69**	13.09	11.04
21DF	3.26	3.20	2.97	**2.26**
InTheWild	5.95	5.98	6.33	**4.32**

5. Conclusion

This paper presents a Rhythm and Speaker Perturbation Copy-Synthesis method for enhancing the performance of anti-spoofing models against various TTS and VC attacks. We analyze the limitations of the copy-synthesis method and locate the artifacts carried by acoustic models of TTS and VC algorithms. By introducing rhythm and speaker perturbation modules, the proposed method successfully guides the spoofing speech detection models to leverage the artifacts introduced by the acoustic models of attack algorithms. Anti-spoofing countermeasures trained with the proposed method achieve performance improvement on multiple evaluation datasets. Our future work will focus on exploring how to extract features related to rhythm and speaker information and utilize them for the detection of spoofing utterances.

6. Acknowledgements

This work is partially supported by the National Key Research and Development Program of China (No.2021YFC3320103).

7. References

[1] V. Shchemelinin and K. Simonchik, "Examining vulnerability of voice verification systems to spoofing attacks by means of a tts system," in *International Conference on Speech and Computer*. Springer, 2013, pp. 132–137.

[2] T. Kinnunen, Z.-Z. Wu, K. A. Lee, F. Sedlak, E. S. Chng, and H. Li, "Vulnerability of speaker verification systems against voice conversion spoofing attacks: The case of telephone speech," in *Proc. ICASSP 2012*. IEEE, 2012, pp. 4401–4404.

[3] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, "Spoofing and countermeasures for speaker verification: A survey," *speech communication*, vol. 66, pp. 130–153, 2015.

[4] 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.

[5] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A comparison of features for synthetic speech detection," in *Proc. Interspeech 2015*, 2015, pp. 2087–2091.

[6] Y. Zhang, W. Wang, and P. Zhang, "The Effect of Silence and Dual-Band Fusion in Anti-Spoofing System," in *Proc. Interspeech 2021*, 2021, pp. 4279–4283.

[7] Z. Wu, R. K. Das, J. Yang, and H. Li, "Light Convolutional Neural Network with Feature Genuinization for Detection of Synthetic Speech Attacks," in *Proc. Interspeech 2020*, 2020, pp. 1101–1105.

[8] J. Monteiro, J. Alam, and T. H. Falk, "Generalized end-to-end detection of spoofing attacks to automatic speaker recognizers," *Computer Speech & Language*, vol. 63, p. 101096, 2020.

[9] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-end anti-spoofing with rawnet2," in *Proc. ICASSP 2021*. IEEE, 2021, pp. 6369–6373.

[10] J.-w. Jung, H.-S. Heo, H. Tak, H.-j. Shim, J. S. Chung, B.-J. Lee, H.-J. Yu, and N. Evans, "Aasist: Audio anti-spoofing using integrated spectro-temporal graph attention networks," in *Proc. ICASSP 2022*. IEEE, 2022, pp. 6367–6371.

[11] X. Wang and J. Yamagishi, "Investigating active-learning-based training data selection for speech spoofing countermeasure," in *2022 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2023, pp. 585–592.

[12] H. Tak, M. Todisco, X. Wang, J. weon Jung, J. Yamagishi, and N. Evans, "Automatic Speaker Verification Spoofing and Deepfake Detection Using Wav2vec 2.0 and Data Augmentation," in *Proc. Odyssey 2022*, 2022, pp. 112–119.

[13] H. Tak, M. Kamble, J. Patino, M. Todisco, and N. Evans, "Rawboost: A raw data boosting and augmentation method applied to automatic speaker verification anti-spoofing," in *Proc. ICASSP 2022*, 2022, pp. 6382–6386.

[14] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *Interspeech 2019*, Sep 2019. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2680

[15] Y. Zhang, F. Jiang, and Z. Duan, "One-class learning towards synthetic voice spoofing detection," *IEEE Signal Processing Letters*, vol. 28, pp. 937–941, 2021.

[16] F. Alegre, A. Amehraye, and N. Evans, "A one-class classification approach to generalised speaker verification spoofing countermeasures using local binary patterns," in *2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, 2013, pp. 1–8.

[17] J. Lu, Y. Zhang, W. Wang, Z. Shang, and P. Zhang, "One-class knowledge distillation for spoofing speech detection," in *Proc. ICASSP 2024*, 2024.

[18] X. Wang and J. Yamagishi, "Spoofed training data for speech spoofing countermeasure can be efficiently created using neural vocoders," in *Proc. ICASSP 2023*, 2023, pp. 1–5.

[19] J. Lu, Y. Zhang, Z. Li, Z. Shang, W. Wang, and P. Zhang, "Detecting unknown speech spoofing algorithms with nearest neighbors," in *Proceedings of IJCAI 2023 Workshop on Deepfake Audio Detection and Analysis*, 2023.

[20] X. Yan, J. Yi, J. Tao, C. Wang, H. Ma, T. Wang, S. Wang, and R. Fu, "An initial investigation for detecting vocoder fingerprints of fake audio," in *Proceedings of the 1st International Workshop on Deepfake Detection for Audio Multimedia*, 2022, pp. 61–68.

[21] Z. Shang, H. Zhang, P. Zhang, L. Wang, and T. Li, "Analysis and solution to aliasing artifacts in neural waveform generation models," *Applied Acoustics*, vol. 203, p. 109183, 2023.

[22] K. Qian, Y. Zhang, S. Chang, M. Hasegawa-Johnson, and D. Cox, "Unsupervised speech decomposition via triple information bottleneck," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, H. D. III and A. Singh, Eds., vol. 119. PMLR, 13–18 Jul 2020, pp. 7836–7846. [Online]. Available: https://proceedings.mlr.press/v119/qian20a.html

[23] D. Cai, Z. Cai, and M. Li, "Identifying source speakers for voice conversion based spoofing attacks on speaker verification systems," in *Proc. ICASSP 2023*, 2023, pp. 1–5.

[24] S. E. McAdams, *Spectral fusion, spectral parsing and the formation of auditory images*. Stanford university, 1984.

[25] J. Patino, N. Tomashenko, M. Todisco, A. Nautsch, and N. Evans, "Speaker Anonymisation Using the McAdams Coefficient," in *Proc. Interspeech 2021*, 2021, pp. 1099–1103.

[26] X. Wang and J. Yamagishi, "Spoofed training data for speech spoofing countermeasure can be efficiently created using neural vocoders," in *Proc. ICASSP 2023*, 2023, pp. 1–5.

[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] X. Liu, X. Wang, M. Sahidullah, J. Patino, H. Delgado, T. Kinnunen, M. Todisco, J. Yamagishi, N. Evans, A. Nautsch, and K. A. Lee, "Asvspoof 2021: Towards spoofed and deepfake speech detection in the wild," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 2507–2522, 2023.

[30] N. Müller, P. Czempin, F. Diekmann, A. Froghyar, and K. Böttinger, "Does Audio Deepfake Detection Generalize?" in *Proc. Interspeech 2022*, 2022, pp. 2783–2787.

[31] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, vol. 33, pp. 12 449–12 460, 2020.

[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.