Robust audio anti-spoofing countermeasure with joint training of front-end and back-end models

Xingming Wang[1,2], *Bang Zeng*[1,2], *Hongbin Suo*[3], *Yulong Wan*[3], *Ming Li*[1,2]

[1]School of Computer Science, Wuhan University, Wuhan, China
[2]Data Science Research Center, Duke Kunshan University, Kunshan, China
[3]Data & AI Engineering System, OPPO, Beijing, China

xingming.wang@dukunshan.edu.cn, ming.li369@dukekunshan.edu.cn

Abstract

The accuracy and reliability of many speech processing systems may deteriorate under noisy conditions. This paper discusses robust audio anti-spoofing countermeasure for audio in noisy environments. Firstly, we attempt to use a pre-trained speech enhancement model as the front-end module and build a cascaded system. However, the independent denoising process of enhancement models may distort the synthesis artifacts or anti-spoofing related information included in utterances, leading to performance degradation. Therefore, we proposes a new framework for robust audio anti-spoofing by joint training the integrated speech enhancement front-end and anti-spoofing back-end. The final results demonstrate that the joint training framework is more effective than the cascaded framework. Additionally, we propose a cross-joint training scheme, which allows the single-model performance to exceed the result of score level fusion, making the joint framework more effective and efficient.

Index Terms: Anti-spoofing, Speaker verification, Spoofing Countermeasure, Speech enhancement

1. Introduction

With the development of deep learning, the performance of both text-to-speech (TTS) and speech conversion (VC) systems has improved significantly in recent years [1]. As a result, human users and Automatic Speaker Verification(ASV) systems are potentially facing increasingly severe threats and security concerns [2]. The synthesized audio anti-spoofing countermeasure system is typically used to detect synthesized spoofing audios, thus enhancing the robustness of ASV systems[3]. The performance of synthesized audio anti-spoofing systems has improved dramatically due to the development of deep learning[4] and numerous new corpora [5, 6, 7]. Most existing works explore the generalization capability of spoof detection systems to unseen synthesizers [8]. In contrast, only little work focuses on the performance of robust audio anti-spoofing systems in noisy scenarios [9, 7, 10]. Considering that realistic applications are usually complex, it motivates us to explore the performance of the audio anti-spoofing system in noisy scenarios.

Background noise usually degrades speech intelligibility and quality, leading to reduced performance of the downstream speech processing tasks. Speech Enhancement (SE) aims to convert the noisy audio into a clean version improving the quality and intelligibility of the original speech. The performance of many speech-to-text tasks, such as automatic speech recognition (ASR) [11, 12, 13] and keyword spotting [14], can be improved with independent pretrained speech enhancement front-end modules. However, some research works show that the pre-trained SE module may distort the speaker information for speaker verification in noisy scenarios [15, 16]. Therefore, Gao et al. [17] use a UNet-DenseNet joint training scheme to address this problem. Kim et al. [18] propose a structure named Extended U-Net for joint training. For noisy audio anti-spoofing countermeasure, Yu et al. [19] uses traditional speech enhancement methods to act on the downstream audio anti-spoofing backends, and Hanilci et al. [20] explores the performance variation of a audio anti-spoofing system in various types of noise scenarios, but all of these works still uses traditional Gaussian Mixture Model (GMM)-based solutions. Ma et al. [10] publish a new dataset containing synthesized audio with noise and use LFCC-LCNN [6] and RawNet2 [21] as the anti-spoofing baseline. However, the authors only use noisy data to train the ordinary anti-spoofing network and do not perform model-level optimization considering noisy scenarios.

In this paper, we first discuss the impact of the speech enhancement front-end module on the performance of audio anti-spoofing back-end systems. Moreover, we propose a new framework for robust audio anti-spoofing with joint training of speech enhancement front-end and audio anti-spoofing back-end models. We use U-Net [22] as the speech enhancement front-end, which has been widely used for speech enhancement task [23]. For the back-end audio anti-spoofing countermeasure module, we explore light convolutional neural network (LCNN) [24] and ResNet [25] architectures. For noisy data, we use three different types of noise in the MUSAN dataset [26] to generate noisy audios under different Signal-to-Noise Ratio (SNR) levels. Our experimental results show that the proposed joint training framework is robust against noisy data, especially for scenarios with low SNR.

In addition, we propose a Cross-Joint Training (CJT) scheme, which aims to combine the different back-end models' information in the integrated model. The CJT scheme is a two-stage training strategy. In the first stage, we train a joint front-end and back-end framework. In the second stage, we keep the front-end model, replace the back-end model with a different structure, and continue training the new joint framework as the final framework. The final results show that this strategy can effectively improve the performance for robust audio anti-spoofing scenarios, even exceeding the results of score level fusion of different models. We additionally use a Voice Activate Detection (VAD) model to trim the ASVSpoof2019 LA data and only use speech segments for training and testing to prove the effectiveness of our system.

Corresponding Author: Ming Li. This research is funded in part by the National Natural Science Foundation of China (62171207)Science and Technology Program of Suzhou City(SYC2022051) and OPPO. Many thanks for the computational resource provided by the Advanced Computing East China Sub-Center.

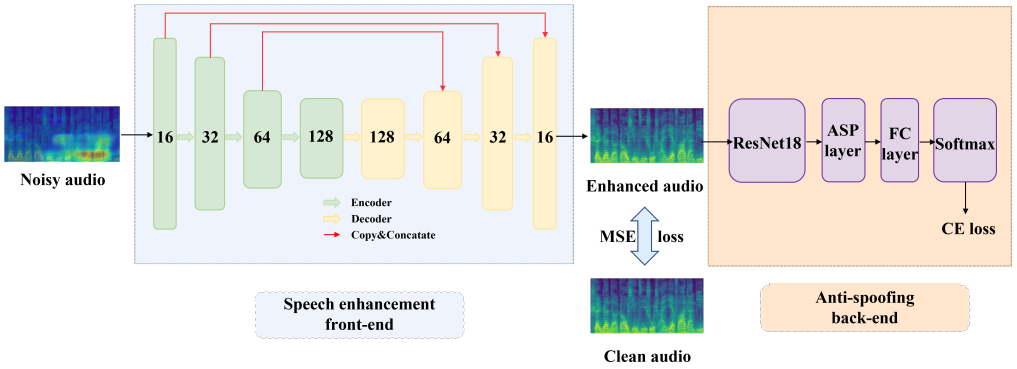

Figure 1: *The illustration of the joint training framework.*

2. Methods

2.1. Speech Enhancement Front-end

The U-Net architecture is widely used for speech processing, such as speech enhancement. It is a symmetrical encoder-decoder architecture. The encoder is composed of convolutional layers, and the decoder is composed of transposed convolutional layers, making the output features the same size as its input. The fundamental idea of using U-Net as a speech enhancement module is to predict a clean spectrogram of noisy speech. For a pair of clean and noisy spectrograms as $C(k,l)$ and $N(k,l)$, where k denotes index into the frequency bins, and l is for the time frame, U-Net takes $N(k,l)$ as the input and predicts $\hat{N}(k,l)$, the recovered spectrogram. The loss L_{mse} is designed to minimize the mean squared error (MSE) between the clean spectrogram and the recovered spectrogram.

$$\mathcal{L}_{mse} = \sum_{k}\sum_{l} ||\hat{N}(k,l) - C(k,l)||^2$$

2.2. Audio Anti-spoofing Back-end

The goal of audio anti-spoofing countermeasure is to identify synthetic audio. We use two different network structures as the audio anti-spoofing back-ends, LCNN [24] and ResNet18 [25]. Both are commonly used for audio anti-spoofing in ASVSpoof 2021 challenge [6]. The Max-Feature-Map (MFM) operation based on the Max-Out activation function is the essential component in the LCNN. The Bi-LSTM layer is used for pooling to aggregate utterance-level embeddings in LCNN. For ResNet18, which is a light-weight version of ResNet, we use Attentive Statistic Pooling (ASP) [14] blocks to aggregate utterance-level embeddings. We use the binary cross-entropy loss as follows:

$$\mathcal{L}_{ce} = \sum_{i} -(y_i \log(p_i) + (1-y_i)\log(1-p_i))$$

where $y_i \in \{0,1\}$ is class label and p_i is the probability output of the classifier.

2.3. Method 1: Fixed pre-trained SE front-end

If the speech enhancement (SE) and the audio anti-spoofing modules were optimized independently, both the training and testing data of the anti-spoofing model should have gone through the speech enhancement front-end. Therefore, we pre-train a U-Net speech enhancement model and freeze it. The enhanced spectrogram of the fixed pre-trained U-net model is used as the input for training and testing the downstream audio anti-spoofing module. This method can be seen as a cascaded anti-spoofing system using a frozen speech enhancement front-end.

2.4. Method 2: Joint training framework

Although using the enhanced spectrogram of the SE module to train downstream audio anti-spoofing network as mentioned in 2.3 may be feasible, the SE module is only designed to recover the clean spectrogram. It does not consider information related to anti-spoofing detection, such as artifacts originated from speech synthesis systems. It may affect or destroy some spoofing clues in the noisy audio data. In addition, channel and environment differences in the training data of the independently optimized modules may also cause domain mismatch. To address these issues, we propose a new framework for robust audio anti-spoofing using jointly training the integrated speech enhancement front-end and audio anti-spoofing back-end. As shown in Figure 1, the SE module (U-Net) and the audio anti-spoofing module are combined and trained together, and the output spectrogram of U-Net is directly used as the input to the downstream network. We use MSE loss and CE loss together as the combined loss function during the joint training.

$$\mathcal{L} = \mathcal{L}_{ce} + \mathcal{L}_{mse}$$

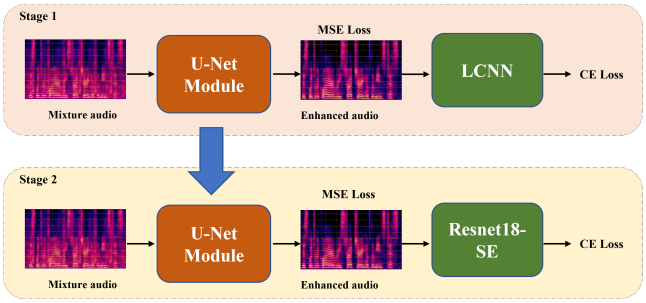

Figure 2: *The illustration of the cross-joint training strategy.*

2.5. Method 3: Cross-Joint Training

As shown in Figure 2, considering the cascade relationship between the front-end and back-end modules in method 2, we attempt to jointly train the U-Net front-end with different back-end modules one by one, which is the proposed Cross-Joint

Training (CJT) approach. For example, in the first stage, we train the **U-Net-LCNN** joint framework until the model converges. In the second stage, we retrain the **U-Net-ResNet** model using the parameters of the U-Net module from the **U-Net-LCNN** model in the first stage. No module is frozen during the whole experiment.

3. Experiments Setup

3.1. Data preparation

Two datasets are used in our experiments, the FAD dataset [10] and the ASVSpoof 2019 LA dataset [5]. For the FAD dataset, we use the training, development and test set as officially provided in [10]. Both clean and noisy data in the training set are used for training. For the ASVSpoof 2019 LA dataset, the MUSAN corpus [26] is employed to generate noisy audio. We divide the MUSAN corpus into two non-overlapping parts for training and testing. The MUSAN corpus contains three different types of noisy data. For the training and development sets of the ASVSpoof 2019 LA dataset, the noisy audio is generated using the training part of the MUSAN corpus. For the evaluation set of the ASVSpoof 2019 LA dataset, the noisy audio is generated using the testing part of the MUSAN corpus. Signal-to-Noise Ratios (SNR) are set to 0 dB, 5 dB, 10 dB, 15 dB, and 20 dB. All noisy audio is generated offline. The Equal Error Rate (EER) is used as the primary metric in subsequent evaluations.

3.2. Model configurations

For the U-Net model, the total number of blocks is set as 8, with 4 blocks in the encoder and 4 blocks in the decoder. The number of channels for each layer in the encoder is set as 16, 32, 64, and 128. Besides, the U-Net has one convolution layer and one transposed convolution layer. A more detailed architecture description can be found in Table 1.

For the LCNN and ResNet18 models, the embedding size is set to 256. Squeeze-and-Excitation block [27] is also used in the ResNet18 model, and the dimension of the bottleneck in the Squeeze-and-Excitation block is set to 256.

3.3. Training configurations

For feature extraction, the logarithmical Mel-spectrogram is extracted by applying 64 Mel filters on the spectrogram computed over Hamming windows of 25ms shifted by 10ms. For joint training, the learning rate is set as 1e-3 during training. For CJT, the learning rate is set as 1e-3 during stage 1 while 1e-4 during stage 2. We adopt the Reduceonplateau learning rate (LR) scheduler with 0.1 initial LR. All models are trained using the Adam optimizer.

4. Results and discussion

This section presents our experimental results and analysis. We firstly discuss whether a fixed pre-trained SE U-Net module can help with downstream audio anti-spoofing. And then we demonstrate and analyze the effectiveness and robustness of the proposed joint training framework and the CJT strategy.

4.1. Does a separate pre-trained SE module work for downstream audio anti-spoofing in noisy conditions?

To verify the impact of the pre-trained speech enhancement module on the downstream audio anti-spoofing task, we conducted experiments as follows. Initially, we pre-trained a U-Net speech enhancement model using both the noisy and clean version of the training and development set of the ASVSpoof 2019 LA dataset. The pre-trained U-Net model was then frozen, and the enhanced spectrogram was used as input for the downstream LCNN module, which was optimized independently using only CE loss. This downstream back-end was considered as a separate training anti-spoofing module, named ***U-Net(fixed)+LCNN*** in Table 3. We compared the performance of this cascaded method with training the original audio anti-spoofing back-end using all noisy and clean audio directly as input. The results can be shown in rows 1-5 of Table 3.

The results showed that for the 0dB SNR scenario, even the ***AASIST*** [28], which performed well on the naive ASVSpoof2019 LA dataset [5], exhibited severe performance degradation. Although the pre-trained U-Net was involved in the training process of the downstream LCNN module, no significant improvement can be observed. Furthermore, using pre-trained U-Net as the front-end of the ***U-Net(fixed)+ResNet18*** model led to performance degradation, indicating that the independently optimized method may not be suitable for the noisy audio anti-spoofing task.

Table 1: *The U-Net based speech enhancement network architecture.* $\boldsymbol{C}(kernel, stride, channel)$ *denotes the 2D convolution layer while* $\boldsymbol{TC}(kernel, stride, channel)$ *denotes the 2D transposed convolution layer.* ***SE*** *denotes the Squeeze-and-Excitation block here. [27]* ***EBx*** *denotes corresponding encoder block x.* $[\cdot]$ *denotes the basic block.*

Layer name	Layer Structure	Output size
Conv1	**C**(7,1,16)	(16,32,T)
Encoder Block 1	$\begin{bmatrix}\mathbf{C}(3,1,16)\\ \mathbf{C}(3,1,32)\\ \mathbf{SE}\end{bmatrix} \times 3$	(16,32,T)
Encoder Block 2	$\begin{bmatrix}\mathbf{C}(3,2,32)\\ \mathbf{C}(3,1,32)\\ \mathbf{SE}\end{bmatrix} \times 4$	(32,32,$\frac{T}{2}$)
Encoder Block 3	$\begin{bmatrix}\mathbf{C}(3,2,64)\\ \mathbf{C}(3,1,64)\\ \mathbf{SE}\end{bmatrix} \times 6$	(64,8,$\frac{T}{4}$)
Encoder Block 4	$\begin{bmatrix}\mathbf{C}(3,1,128)\\ \mathbf{C}(3,1,128)\\ \mathbf{SE}\end{bmatrix} \times 3$	(128,8,$\frac{T}{4}$)
Decoder Block 1	$\begin{bmatrix}ConcatateEB4\\ \mathbf{C}(3,1,32)\end{bmatrix}$	(64,8,$\frac{T}{4}$))
Decoder Block 2	$\begin{bmatrix}ConcatateEB3\\ \mathbf{TC}(2,1,64)\end{bmatrix}$	(32,16,$\frac{T}{2}$)
Decoder Block 3	$\begin{bmatrix}ConcatateEB2\\ \mathbf{TC}(2,1,128)\end{bmatrix}$	(16,32,T)
Decoder Block 4	$\begin{bmatrix}ConcatateEB1\\ \mathbf{C}(1,1,256)\end{bmatrix}$	(16,32,T)
Transpose Conv2	**TC**(2X1,2X1,1)	(1,32,T)

4.2. Results of the joint training framework

Table 2 shows the performance improvement of our proposed joint training framework on the ASVSpoof 2019 LA noisy evaluation set under different SNR scenarios. As can be seen from the table, the joint training model brings considerable improvement in different SNR scenarios compared to training the anti-

Table 2: *System performance of the joint training framework and the anti-spoofing only system on the ASVSpoof2019 LA noisy evaluation set. The metric used is EER (%).*

Model	noise					babble					music				
	0dB	5dB	10dB	15dB	20dB	0dB	5dB	10dB	15dB	20dB	0dB	5dB	10dB	15dB	20dB
LCNN	15.62	11.21	10.59	11.00	6.45	17.73	13.00	12.33	8.68	5.69	15.08	13.00	10.02	7.59	6.56
U-Net-LCNN	**9.52**	8.59	**7.09**	6.52	4.13	**9.30**	**6.28**	8.56	6.10	3.91	10.63	8.24	6.79	5.02	4.48
ResNet18	15.07	8.85	8.42	8.02	6.95	17.47	9.41	9.46	8.63	6.81	16.52	10.79	7.20	6.25	4.98
U-Net-ResNet18	9.99	**5.93**	7.45	**5.37**	**4.00**	10.69	8.02	**7.39**	**3.16**	**3.89**	**10.41**	**6.02**	**6.48**	**3.59**	**3.95**

Table 3: *System performance on the ASVSpoof2019 LA noisy evaluation set. The metric used is EER(%).*

ID	Model	noise-0dB	babble-0dB	music-0dB
1	*LCNN*	15.62	17.73	15.08
2	*ResNet18*	15.07	17.47	16.52
3	*AASIST*	15.04	15.34	14.07
4	*U-Net(fixed) + LCNN*	14.60	16.33	14.83
5	*U-Net(fixed)+ResNet18*	14.52	18.32	15.39
6	*U-Net-LCNN*	9.52	9.30	10.63
7	*U-Net-ResNet18*	9.99	10.69	10.41
8	*U-Net-LCNN(CJT)*	8.07	8.71	8.49
9	*U-Net-ResNet18(CJT)*	7.99	**7.76**	**7.18**
10	*6+7(score fuse)*	**8.06**	8.31	9.37

Table 4: *System performance of the joint training framework and the anti-spoofing only system on the FAD test set. The method in* Baseline*[10] is* LFCC-LCNN.

Model	noisy		clean	
	seen	unseen	seen	unseen
Baseline[10]	6.88	29.67	1.26	26.56
AASIST	2.25	25.94	0.882	26.32
LCNN	1.48	29.72	0.61	26.76
ResNet18	1.65	28.08	0.69	**24.46**
U-Net-LCNN	1.27	27.11	0.67	28.74
U-Net-ResNet18	**1.06**	**25.76**	**0.45**	25.46

spoofing only system using only all noisy and clean audio. Especially for low SNR scenarios, for noise audio with 0dB SNR, the ***U-Net-LCNN*** can reduce the EER by 50% compared to the ***LCNN***. The ***Unet-ResNet18*** model is relatively more stable for music-type noise. The performance of ordinary anti-spoofing models for babble-type noise is relatively poor compared with the other two types of noise. This decline may be due to the inclusion of bonafide audio in the babble noise data, making the anti-spoofing task more difficult.

Table 4 presents the results on the FAD official test set. Compared with the baseline and the ***AASIST*** system, our proposed joint training model performs better on the noisy test set, regardless of whether the scenario is seen or unseen. For the clean test set, it can be observed that our joint framework may suffer from slight performance degradation for the unseen data. Given the models' superior performance on the noisy test set, we believe it is acceptable.

4.3. Results of the cross-joint training strategy

The effectiveness of our proposed CJT strategy can be demonstrated in rows 6-10 of Table 3. For the ***U-Net-ResNet18(CJT)*** model, we load the U-Net model from the ***U-Net-LCNN*** model as stage 1. As can be seen from the final results, the model trained with the CJT strategy even outperformes the score level fusion approach with babble and music noises. We speculate that this cross-training approach can introduce information from differently structured models, thereby improving the performance of the final model. The CJT strategy makes the final model not only perform well but also more efficient during inference since there is only one back-end model.

Table 5: *System performance of the joint training framework and anti-spoofing only system on the ASVSpoof2019 LA noisy evaluation set after VAD.*

Model	noise-0dB	babble-0dB	music-0dB
LCNN	25.09	21.39	21.76
ResNet18	21.35	20.20	23.22
U-Net-LCNN	24.16	**19.05**	20.53
U-Net-ResNet18	**19.21**	19.81	**19.63**

4.4. Results after Voice Activate Detection (VAD)

As mentioned in [29], there is controversy over the ASVSpoof2019 dataset containing silent segments that can aid in spoofing audio detection. Therefore, we further use a VAD model to trim the ASVSpoof2019 LA data and only use speech segments for training and testing. We use a conformer-based VAD model as used in [30]. The final results are shown in Table 5. It can be seen that compared to the results in Table 3, the performance of the audio anti-spoofing countermeasure system does degrade significantly when only using the speech segments. However, our proposed joint training model remains effective under this condition.

5. Conclusion

This paper aims to build a robust anti-spoofing countermeasure system for audio signals in noisy environments. To achieve this, we initially attempted to utilize a pre-trained speech enhancement U-Net module as the front-end to build a cascaded framework. The results show that this approach does not necessarily improve performance and even degrade the performance in certain conditions. Therefore, we propose a novel framework for robust audio anti-spoofing by jointly training an integrated speech enhancement front-end and anti-spoofing back-end model. Experimental results demonstrate that the joint training framework is more effective than both the anti-spoofing only system and the separate training cascaded framework. In addition, we also propose a cross-joint training method to further enhance performance in low signal-to-noise ratio scenarios.

6. References

[1] X. Tan, T. Qin, F. Soong, and T.-Y. Liu, "A survey on neural speech synthesis," *arXiv preprint arXiv:2106.15561*, 2021.

[2] N. W. Evans, T. Kinnunen, and J. Yamagishi, "Spoofing and countermeasures for automatic speaker verification." in *Proc. Interspeech*, 2013, pp. 925–929.

[3] Z. Bai and X.-L. Zhang, "Speaker recognition based on deep learning: An overview," *Neural Networks*, vol. 140, pp. 65–99, 2021.

[4] 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.

[5] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. Evans, T. H. Kinnunen, and K. A. Lee, "ASVspoof 2019: Future Horizons in Spoofed and Fake Audio Detection," in *Proc. Interspeech 2019*, pp. 1008–1012.

[6] J. Yamagishi, X. Wang, M. Todisco, M. Sahidullah, J. Patino, A. Nautsch, X. Liu, K. A. Lee, T. Kinnunen, N. Evans, and H. Delgado, "ASVspoof 2021: accelerating progress in spoofed and deepfake speech detection," in *Proc. ASVspoof2021 workshop*, 2021, pp. 47–54.

[7] J. Yi, R. Fu, J. Tao, S. Nie, H. Ma, C. Wang, T. Wang, Z. Tian, Y. Bai, C. Fan *et al.*, "Add 2022: the first audio deep synthesis detection challenge," in *Proc. ICASSP 2022*, pp. 9216–9220.

[8] X. Liu, X. Wang, M. Sahidullah, J. Patino, H. Delgado, T. Kinnunen, M. Todisco, J. Yamagishi, N. Evans, A. Nautsch *et al.*, "Asvspoof 2021: Towards spoofed and deepfake speech detection in the wild," *arXiv preprint arXiv:2210.02437*.

[9] X. Tian, Z. Wu, X. Xiao, E. S. Chng, and H. Li, "An Investigation of Spoofing Speech Detection Under Additive Noise and Reverberant Conditions," in *Proc. Interspeech 2016*, 2016, pp. 1715–1719.

[10] H. Ma, J. Yi, C. Wang, X. Yan, J. Tao, T. Wang, S. Wang, L. Xu, and R. Fu, "Fad: A chinese dataset for fake audio detection," *arXiv preprint arXiv:2207.12308*, 2022.

[11] C. Donahue, B. Li, and R. Prabhavalkar, "Exploring speech enhancement with generative adversarial networks for robust speech recognition," in *Proc. ICASSP 2018*, pp. 5024–5028.

[12] A. Pandey, C. Liu, Y. Wang, and Y. Saraf, "Dual application of speech enhancement for automatic speech recognition," in *Proc. SLT 2021*, pp. 223–228.

[13] K. Iwamoto, T. Ochiai, M. Delcroix, R. Ikeshita, H. Sato, S. Araki, and S. Katagiri, "How bad are artifacts?: Analyzing the impact of speech enhancement errors on ASR," in *Proc. Interspeech 2022*, pp. 5418–5422.

[14] M. Yu, X. Ji, B. Wu, D. Su, and D. Yu, "End-to-End Multi-Look Keyword Spotting," in *Proc. Interspeech 2020*, pp. 66–70.

[15] D. Cai, W. Cai, and M. Li, "Within-sample variability-invariant loss for robust speaker recognition under noisy environments," in *Proc. ICASSP 2020*, pp. 6469–6473.

[16] S. Shon, H. Tang, and J. Glass, "VoiceID Loss: Speech Enhancement for Speaker Verification," in *Proc. Interspeech 2019*, pp. 2888–2892.

[17] Z. Gao, M. Mak, and W. Lin, "UNet-DenseNet for Robust Far-Field Speaker Verification," in *Proc. Interspeech 2022*, pp. 3714–3718.

[18] J.-H. Kim, J. Heo, H. jin Shim, and H.-J. Yu, "Extended U-Net for Speaker Verification in Noisy Environments," in *Proc. Interspeech 2022*, pp. 590–594.

[19] H. Yu, A. Sarkar, D. A. L. Thomsen, Z.-H. Tan, Z. Ma, and J. Guo, "Effect of multi-condition training and speech enhancement methods on spoofing detection," in *2016 First International Workshop on Sensing, Processing and Learning for Intelligent Machines (SPLINE)*. IEEE, 2016, pp. 1–5.

[20] C. Hanilci, T. Kinnunen, M. Sahidullah, and A. Sizov, "Spoofing detection goes noisy: An analysis of synthetic speech detection in the presence of additive noise," *Speech Communication*, vol. 85, pp. 83–97, 2016.

[21] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-End anti-spoofing with RawNet2," in *Proc. ICASSP 2021*, pp. 6369–6373.

[22] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI*, 2015, pp. 234–241.

[23] H.-S. Choi, J.-H. Kim, J. Huh, A. Kim, J.-W. Ha, and K. Lee, "Phase-aware speech enhancement with deep complex u-net," in *Proc. ICLR*, 2018.

[24] X. Wu, R. He, Z. Sun, and T. Tan, "A light cnn for deep face representation with noisy labels," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2884–2896, 2018.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[26] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," *arXiv:1510.08484*.

[27] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. CVPR*, 2018, pp. 7132–7141.

[28] 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, pp. 6367–6371.

[29] N. Müller, F. Dieckmann, P. Czempin, R. Canals, K. Böttinger, and J. Williams, "Speech is Silver, Silence is Golden: What do ASVspoof-trained Models Really Learn?" in *Proc. ASVspoof2021 workshop*, pp. 55–60.

[30] W. Wang, X. Qin, M. Cheng, Y. Zhang, K. Wang, and M. Li, "The dku-dukeece diarization system for the voxceleb speaker recognition challenge 2022," *arXiv preprint arXiv:2210.01677*.