Interspeech 2025
17-21 August 2025, Rotterdam, The Netherlands

Amplifying Artifacts with Speech Enhancement in Voice Anti-spoofing

Thanapat Trachu[*,1], *Thanathai Lertpetchpun*[*,†,2], *Ekapol Chuangsuwanich*[1]

[1]Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Thailand
[2]University of Southern California, Los Angeles, USA

thanapat.trachu@gmail.com, lertpetc@usc.edu, ekapol.c@chula.ac.th

Abstract

Spoofed utterances always contain artifacts introduced by generative models. While several countermeasures have been proposed to detect spoofed utterances, most primarily focus on architectural improvements. In this work, we investigate how artifacts remain hidden in spoofed speech and how to enhance their presence. We propose a model-agnostic pipeline that amplifies artifacts using speech enhancement and various types of noise. Our approach consists of three key steps: noise addition, noise extraction, and noise amplification. First, we introduce noise into the raw speech. Then, we apply speech enhancement to extract the entangled noise and artifacts. Finally, we amplify these extracted features. Moreover, our pipeline is compatible with different speech enhancement models and countermeasure architectures. Our method improves spoof detection performance by up to 44.44% on ASVspoof2019 and 26.34% on ASVspoof2021.

Index Terms: Voice Anti-spoofing, Speech Enhancement, Features Amplification

1. Introduction

The voice anti-spoofing task involves classifying whether an input utterance is spoofed or bona fide. To protect automatic speaker verification (ASV) systems from spoofing attacks, a separate system called a countermeasure (CM) is typically developed to detect spoofed utterances. In the ASVspoof2019 challenge [1], two main tasks were introduced: physical attacks (PA) and logical attacks (LA). PA focuses on defending against replay attacks, while LA targets speech generated by text-to-speech (TTS) and voice conversion (VC) systems. The ASVspoof2021 challenge [2] introduced codecs to increase the difficulty of defending against attacks. [3] Deng et al. proposed a pipeline to detect spoofed voices using vocoder fingerprints, indicating that different VCs have different artifacts. In ASVspoof2024 [4], adversarial attacks were included alongside new TTS and VC architectures to ensure countermeasures remain robust against evolving spoofing techniques. In this work, we focus on logical attacks, where utterances are generated using text-to-speech and voice conversion models.

Speech enhancement aims to remove noise from a noisy speech signal to produce a cleaner output. It is commonly used alongside other speech processing tasks, such as speech recognition [5, 6, 7] and speaker verification [8], to enhance system robustness against noise. However, as shown in [9], speech enhancement does not solely extract noise but may also remove unrelated information, including parts of the clean speech signal. Given this behavior, we expect that applying speech enhancement to spoofed utterances could similarly remove artifacts.

Wang et al. [10] used speech enhancement for noise reduction in anti-spoofing. In contrast, we leverage speech enhancement as an artifact enhancer to improve countermeasure performance against spoofing attacks. Our approach involves adding noise to the input, extracting the entangled noise and artifacts using speech enhancement, and then amplifying both before reinserting them into the utterance. As a result, spoofed speech exhibits a higher artifact magnitude, while bona fide speech behaves like typical noisy speech.

The proposed pipeline was evaluated on the ASVspoof2019 and ASVspoof2021 datasets. Our results show that it improves countermeasure performance by 44.44% relative improvement and remains robust across different speech enhancement and countermeasure architectures. Additionally, our findings suggest that artifacts located in lower frequency bands are more important than that in higher frequency bands for detecting spoof.

2. Background on CM and SE

2.1. Countermeasures

Countermeasure (CM) systems are designed to determine whether an input utterance is bona fide or spoofed. Various approaches exist for developing countermeasures, including using mel spectrograms with convolutional neural networks, input waveforms with graph neural networks, or end-to-end architectures. Light Convolutional Neural Network (LCNN) [11] was the baseline for ASVspoof2021 and supports multiple feature inputs, such as LFCC, CQT, FFT, and mel spectrograms. The architecture consists of a long sequence of convolutional layers, Max-Feature-Map (MFM) activation, and batch normalization. AASIST [12] and RawNet2 [13] were introduced as baselines for ASVspoof2024. AASIST was designed to extract spectro-temporal features from the waveform using a Heterogeneous Stacking Graph Attention Layer (HS-GAL) and Max Graph Operations (MGO). RawNet2 employs Sinc filters to transform the waveform into features for the model and primarily consists of residual blocks (batch normalization, leaky ReLU, convolutional layers, pooling layers, and feature map scaling), a Gated Recurrent Unit (GRU), and a classifier. While these models differ in architecture and input features, they achieve comparable performance. In our work, we adopt LCNN, AASIST, and RawNet2 as the main baselines.

2.2. Speech Enhancement Model

Speech enhancement aims to improve the quality and intelligibility of speech by modeling noisy speech as the sum of clean

[*]These authors contributed equally.
[†]Work performed at Chulalongkorn University.

 10.21437/Interspeech.2025-362

speech and noise. There are various approaches to developing speech enhancement, including generative [14, 15] and discriminative [16, 17, 18] methods.

Thunder [14], for example, is a generative model based on the diffusion process, which consists of two stages: the forward process and the reverse process. In the forward process, noise is gradually added to the data until it transforms into a predefined distribution, known as the prior distribution. The reverse process then removes noise from a sample drawn from the prior distribution, eventually restoring the clean speech. To perform the reverse process, a model is required to predict the noise in the input, iteratively removing it to recover the clean speech. This type of model is typically trained using mean squared error (MSE) between the output and the added noise. NCSNPP [18], on the other hand, is a discriminative model optimized with a different objective function: the signal-to-noise ratio (SNR). While both Thunder and NCSNPP are based on U-Net architectures, GaGNet [16] introduces a new architecture that mimics the human ability to focus on both coarse and fine-grained details. GaGNet decomposes speech enhancement into two subtasks: glance and gaze. The glance path suppresses noise in the magnitude domain of the spectrogram, while the gaze path refines the spectrogram in the complex domain. The model is then trained using the MSE loss function in the frequency domain. In our work, we use Thunder, NCSNPP, and GaGNet to demonstrate that our method improves countermeasure performance regardless of the choice of the speech enhancement model.

3. Method

Our proposed approach leverages an off-the-shelf speech enhancement model to amplify artifacts in spoofed utterances. We hypothesize that while speech enhancement models are designed to remove noise, they also eliminate spoofing artifacts present in audio. By extracting and amplifying these artifacts, we aim to make them more distinguishable for the countermeasure system.

An overview of our pipeline is presented in Figure 1. The pipeline consists of three key steps: (1) noise addition, (2) noise extraction, and (3) noise amplification. Finally, the countermeasure model is trained using the amplified utterances. In this section, we provide a detailed explanation of each step.

3.1. Noise Addition

In the first step, we introduce noise into the utterance at a target signal-to-noise ratio (SNR) to simulate noisy speech, as defined by the following equation:

$$y = x + \sqrt{\frac{||x||^2}{||n||^2 \times 10^{\frac{\text{SNR}}{10}}}} \cdot n \tag{1}$$

where $x \in \mathbb{R}^t$ and $n \in \mathbb{R}^t$ represent the raw input utterance and noise signal, respectively, with t denoting the utterance duration (in samples). The operator $||\cdot||$ denotes the L2 norm. The noise signal n can be any type of noise, such as white noise, brown noise, or other environmental noise. The resulting noisy speech is then fed into the speech enhancement model.

3.2. Noise Extraction

We assume that the speech enhancement model not only removes noise n, but also eliminates artifacts presented in spoofed

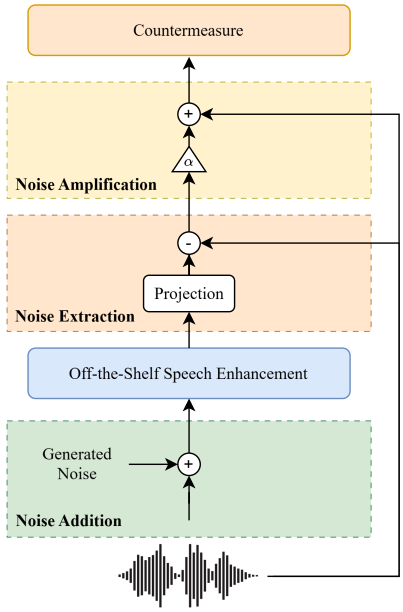

Figure 1: ***Overview of our method.*** *First, noise is added to the input utterances, which are then processed by the speech enhancement model. Noise and artifacts are extracted from the output and reintroduced into the input utterances.*

utterances. We claim that the removed artifacts contain valuable information to help the countermeasure detect spoofed utterances. To leverage this, we introduce a method to extract artifact-contained noise from the output of the speech enhancement model.

In speech enhancement, we typically assume that clean speech and noise are orthogonal. However, [19] demonstrated that directly subtracting clean speech from the estimated clean speech does not guarantee orthogonality. This occurs because the estimated clean speech often has a different scale compared to the target clean speech, leading to a residual clean speech component in the estimated noise. Following the same assumption, we adopt a similar approach as follows:

$$\hat{a} = x - \frac{x \cdot \hat{x}}{||\hat{x}||^2}\hat{x} \tag{2}$$

where $\hat{x}$ is the output of the speech enhancement model, x is the raw input utterance, $\frac{x \cdot \hat{x}}{||\hat{x}||^2}$ is referred to as the *projection weight*, and $\hat{a}$ is the estimated entangled noise and artifact. We hypothesize that the artifact components extracted from spoofed utterances should have a higher magnitude than those from bonafide utterances, as bonafide speech does not contain artifacts. Therefore, speech enhancement acts as an artifact extractor for spoofed speech while functioning as a regular noise reducer for bonafide speech.

3.3. Noise Amplification

In this step, we amplify the extracted noise by adding it back into the raw input utterance x, as shown in the equation below:

$$\tilde{x} = x + \alpha\hat{a} \tag{3}$$

where α is a hyperparameter that controls the degree of noise amplification in the utterance. If the input x is a spoofed utterance, entangled noise and artifacts will be amplified and added back to the speech for all $\alpha > 0$, making it easier for the countermeasure to detect them. In contrast, only the noise will be amplified and added back to bona fide speech. As a result, by amplifying the artifacts, the countermeasure model should be better able to distinguish between bona fide and spoofed utterances. We selected the best alpha based on our grid search parameters, in which $\alpha = 1.4$ for LCNN and RawNet2, and $\alpha = 0.6$ for AASIST.

4. Experimental Setup

4.1. Datasets

4.1.1. VoiceBank + DEMAND

Speech enhancement models were pre-trained on the VoiceBank + DEMAND dataset, consisting of 30 speakers from the VoiceBank Corpus and eight recorded noise samples from DEMAND. The training and validation sets contain 11,572 utterances corrupted by eight recorded noise samples from DEMAND and two artificially generated noise samples (babble and speech-shaped) at SNR levels of 0, 5, 10, and 15 dB, while the testing set contains 824 utterances, each contaminated with five unseen recorded noise samples from DEMAND, at SNR levels of 2.5, 7.5, 12.5, and 17.5 dB. All speech data were sampled at 16 kHz. The training procedure of the speech enhancement models follows Thunder [14].

4.1.2. ASVspoof

We benchmarked the performance of our proposed method on both the ASVspoof2019 [1] and ASVspoof2021 [2] logical access (LA) datasets. The ASVspoof2019LA dataset consists of three subsets: train, development, and evaluation. Spoofed utterances are generated using 19 spoofing attack algorithms (A01-A19), including speech synthesis and voice conversion.

In the training and development sets, spoofed utterances are generated from six different algorithms, while the evaluation set contains spoofed utterances generated by 13 different algorithms. The ASVspoof2021LA dataset, on the other hand, employs the same training and development sets as the ASVspoof2019LA dataset and only introduces a new evaluation set. In this new evaluation set, six different codecs were applied to both bona fide and spoofed utterances from the ASVspoof2021LA evaluation set. We computed the signal-to-noise ratio (SNR) using WADA-SNR [20], and the average SNR for ASVspoof2019 and ASVspoof2021 eval sets is 67.82 and 73.11, which are considered clean speech.

4.2. Training Details and Evaluation Measures

4.2.1. Speech Enhancement

We used the official implementation of Thunder[1] to pre-train the Thunder, NCSNPP, and GaGNet models on the VoiceBank+DEMAND dataset. All models were trained for 100 epochs on an A100 GPU using the Adam optimizer, a learning rate of 2×10^{-5}, and a batch size of 8. For other hyperparameters, we followed the default settings from their respective papers. We then selected the epoch with the lowest validation loss as the pretrained model. No augmentation was used during training.

[1] https://github.com/SLSCU/thunder-speech-enhancement

Table 1: *Performance of different speech enhancement and countermeasure models on the ASVspoof2019 and ASVspoof2021 datasets. "SE" and "CM" denote the speech enhancement and countermeasure models, respectively. The best-performing model in each section is bolded.*

Model		**2019**		**2021**	
SE	CM	t-DCF	EER	t-DCF	EER
-	LCNN	0.136	6.20	0.341	7.21
Thunder	LCNN	**0.102**	**4.46**	**0.310**	5.56
GaGNet	LCNN	0.117	4.84	0.322	5.82
NCSNPP	LCNN	0.106	4.70	0.312	**5.31**
-	AASIST	0.048	1.81	0.370	6.70
Thunder	AASIST	0.043	1.55	**0.339**	**5.87**
GaGNet	AASIST	**0.035**	**1.24**	0.363	6.55
NCSNPP	AASIST	0.057	2.07	0.358	6.57
-	RawNet2	0.110	4.83	0.385	8.10
Thunder	RawNet2	**0.062**	2.68	0.349	6.88
GaGNet	RawNet2	0.065	**2.65**	0.359	7.09
NCSNPP	RawNet2	0.078	3.33	**0.341**	**6.67**

4.2.2. Countermeasures

Our code was mainly based on the ASVspoof2021 baseline[2]. Anti-spoofing countermeasure models were trained for 30 epochs on an A100 GPU, using the Adam optimizer, with a learning rate of $1e-4$ that decayed by 0.9 after 10 epochs and a batch size of 32. White noise were used with the default 0 dB SNR in the noise addition process. The utterances were randomly cropped into 4 seconds for both training and evaluation. We used the tandem detection cost function (t-DCF) [21] and equal error rate (EER) as primary metrics, following the ASVspoof2021 challenge [2]. We did not perform any augmentation method in our pipeline.

5. Results and Discussion

5.1. Main Results

We evaluated the pipeline with multiple combinations of speech enhancement (SE) and countermeasure (CM) models to demonstrate its robustness across different architectures. Both discriminative and generative speech enhancement models work well with CNN-based and GNN-based countermeasures. Table 1 shows that our method is effective across various speech enhancement and countermeasure models, compared to using the countermeasure model alone. Moreover, it generalizes well to both ASVspoof2019 and ASVspoof2021. Performance is relatively improved by up to 45% in EER and 43% in t-DCF on ASVspoof2019, and when evaluated on ASVspoof2021, the pipeline demonstrates a relative performance gain of up to 26% in EER and 11% in t-DCF.

5.2. Impact of Generated Noise

The objective of adding noise is to encourage the speech enhancement model to be more aggressive in removing noise and artifacts, as we observed greater SNR improvements when fed low-SNR inputs. We conducted an experiment on ASVspoof2021 with Thunder, comparing systems with and without the noise addition step. Table 2 shows that adding white

[2] https://github.com/asvspoof-challenge/2021/tree/main/LA/Baseline-RawNet2

noise before speech enhancement helps the model extract artifacts more effectively, achieving better results than without noise addition. Speech enhancement typically generates artifacts that degrade the performance of downstream tasks, as observed in ASR tasks [22]. Without the noise from the noise addition process, the extracted noise would primarily consist of speech enhancement artifacts, making it difficult for the countermeasure model to distinguish between spoofed and bona fide utterances.

Table 2: *Performance of different countermeasures in the absence of white noise on the ASVspoof2021 dataset. The noise addition step is crucial for speech enhancement to remove artifacts from the utterance.*

CM	Noise Addition	t-DCF	EER
LCNN	Without noise	0.352	8.67
LCNN	With noise	**0.310**	**5.56**
AASIST	Without noise	0.356	7.09
AASIST	With noise	**0.339**	**5.87**
RawNet2	Without noise	0.478	10.46
RawNet2	With noise	**0.349**	**6.88**

5.3. Influence of Generated Noise Types

We evaluated our method on ASVspoof2021 using three types of noise: white, violet, and pink noise, each with distinct characteristics. White noise has equal power across all frequency bands, while violet noise has greater intensity at higher frequencies, and pink noise has higher intensity at lower frequencies. Using different types of noise helps the speech enhancement model detect artifacts across various frequency bands. As shown in Table 3, the best performance is achieved when using white noise, suggesting that the artifacts are present across all frequency bands. Furthermore, the model performs worse with both pink noise and violet noise compared to white noise, possibly due to challenges in detecting artifacts in certain frequency bands. Moreover, as suggested in [3], different vocoders create different kinds of artifacts. Our analysis revealed that different spoofing algorithms generate artifacts in different frequency bands. For example, A08 is more easily detected with violet noise, while A18 is better detected with pink noise.

Table 3: *Performance across different noise types. White noise yields the best results. Pink noise outperforms violet noise, indicating artifacts are more prominent in lower frequencies.*

SE	CM	Noise	t-DCF	EER
-	RawNet2	-	0.385	8.10
Thunder	RawNet2	Violet	0.358	7.53
Thunder	RawNet2	Pink	0.351	7.01
Thunder	RawNet2	White	**0.349**	**6.88**

6. Ablation Studies

6.1. Effect of Using Projection Weight

We validated our projection approach in the noise extraction step on the ASVspoof 2021 dataset using Thunder and RawNet2. We compared the performance of scaling the output of the speech enhancement model with projection weights versus using no scaling. Without scaling, we expect that the extracted artifacts will retain residual clean speech components, making them less distinct and hindering the countermeasure model's ability to detect them. As shown in Table 4, our proposed projection method in the noise extraction layer improves performance by approximately 8% compared to the naive approach. This result supports our hypothesis that clean speech components remain in the extracted noise from speech enhancement. Directly using the extracted noise from the naive approach amplifies and reintroduces these clean speech components into the input of the countermeasure model, leading to inferior performance compared to the projection-based method.

Table 4: *Performance across different noise extraction methods. Using projection outperforms the naive approach as it extracts only the artifact information.*

SE	CM	Noise Extraction	t-DCF	EER
-	RawNet2	-	0.385	8.10
Thunder	RawNet2	without Projection	0.378	7.48
Thunder	RawNet2	with Projection	**0.349**	**6.88**

6.2. Effect of the SNR of the Generated Noise

The SNR of the added noise determines the energy ratio between the speech component and the artifact component, affecting the noise extraction and amplification processes. We performed the experiment with different SNR levels and evaluated the performance on both ASVspoof2019 and ASVspoof2021, using Thunder and RawNet2 as the speech enhancement and the countermeasure model.

As shown in Table 5, adding too much or too little noise can be detrimental to the framework's performance. In the former case, excessive noise may obscure the presence of the artifacts as entangled noise and artifacts are amplified by the same α during the noise amplification process, maintaining the energy ratio. When the SNR between noise and artifacts is too high, the model may interpret it as a typical noisy speech. In the latter case, insufficient noise may prevent the speech enhancement model from detecting artifacts in certain frequency bands. Our results indicate that using $SNR = 0$ is the most optimal SNR, which can relatively improve up to 33.82% in EER compared with $SNR = 10$.

Table 5: *Effect of SNR levels in the noise addition step.*

	2019		**2021**	
SNR	t-DCF	EER	t-DCF	EER
-5	0.080	3.75	0.363	7.28
0	**0.062**	**2.68**	**0.349**	**6.88**
5	0.066	2.86	0.379	7.41
10	0.093	4.05	0.386	8.13

7. Conclusion

We propose a pipeline for amplifying artifacts in speech utterances using speech enhancement. The pipeline comprises three steps. First, some noise is added to the raw utterance to simulate noisy speech, prompting the speech enhancement model to remove artifacts. Next, we feed the noisy speech into the speech enhancement model and extract the noise from the output before amplifying the noise back into the clean speech. Finally, we use this noise-amplified data to train the countermeasure model. Our method is compatible with various speech enhancement and countermeasure models and improves countermeasure performance on both the ASVspoof2019 and ASVspoof2021 datasets.

8. References

[1] 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.

[2] 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. Interspeech*, 2021, pp. 47–54.

[3] J. Deng, Y. Ren, T. Zhang, H. Zhu, and Z. Sun, "VFD-Net: Vocoder Fingerprints Detection for Fake Audio," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2024, pp. 12 151–12 155.

[4] X. Wang, H. Delgado, H. Tak, J. weon Jung, H. jin Shim, M. Todisco, I. Kukanov, X. Liu, M. Sahidullah, T. H. Kinnunen, N. Evans, K. A. Lee, and J. Yamagishi, "ASVspoof 5: Crowdsourced Speech Data, Deepfakes, and Adversarial Attacks at Scale," in *The Automatic Speaker Verification Spoofing Countermeasures Workshop*, 2024, pp. 1–8.

[5] Q.-S. Zhu, J. Zhang, Z.-Q. Zhang, and L.-R. Dai, "A Joint Speech Enhancement and Self-Supervised Representation Learning Framework for Noise-Robust Speech Recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 1927–1939, 2023.

[6] A. L. Maas, Q. V. Le, T. M. O'Neil, O. Vinyals, P. Nguyen, and A. Y. Ng, "Recurrent Neural Networks for Noise Reduction in Robust ASR," in *Proc. Interspeech*, 2012, pp. 22–25.

[7] Y. Koizumi, S. Karita, A. Narayanan, S. Panchapagesan, and M. Bacchiani, "SNRi Target Training for Joint Speech Enhancement and Recognition," in *Proc. Interspeech*, 2022, pp. 1173–1177.

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

[9] J.-M. Lemercier, J. Richter, S. Welker, and T. Gerkmann, "StoRM: A Diffusion-Based Stochastic Regeneration Model for Speech Enhancement and Dereverberation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 2724–2737, 2023.

[10] X. Wang, B. Zeng, S. Hongbin, Y. Wan, and M. Li, "Robust Audio Anti-spoofing Countermeasure with Joint Training of Front-end and Back-end Models," in *Proc. Interspeech*, 2023, pp. 4004–4008.

[11] G. Lavrentyeva, S. Novoselov, A. Tseren, M. Volkova, A. Gorlanov, and A. Kozlov, "STC Antispoofing Systems for the ASVspoof2019 Challenge," in *Proc. Interspeech*, 2019, pp. 1033–1037.

[12] 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 *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022, pp. 6367–6371.

[13] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-End Anti-spoofing with RawNet2," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 6369–6373.

[14] T. Trachu, C. Piansaddhayanon, and E. Chuangsuwanich, "Thunder : Unified Regression-Diffusion Speech Enhancement with a Single Reverse Step using Brownian Bridge," in *Proc. Interspeech*, 2024, pp. 1180–1184.

[15] J. Richter, S. Welker, J.-M. Lemercier, B. Lay, and T. Gerkmann, "Speech Enhancement and Dereverberation With Diffusion-Based Generative Models," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 2351–2364, 2023.

[16] A. Li, C. Zheng, L. Zhang, and X. Li, "Glance and Gaze: A Collaborative Learning Framework for Single-channel Speech Enhancement," *Applied Acoustics*, vol. 187, p. 108499, 2022.

[17] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing Ideal Time–Frequency Magnitude Masking for Speech Separation," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 27, no. 8, p. 1256–1266, Aug. 2019.

[18] J.-M. Lemercier, J. Richter, S. Welker, and T. Gerkmann, "Analysing Diffusion-based Generative Approaches Versus Discriminative Approaches for Speech Restoration," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2023, pp. 1–5.

[19] J. L. Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "SDR – Half-baked or Well Done?" in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 626–630.

[20] C. Kim and R. M. Stern, "Robust signal-to-noise ratio estimation based on waveform amplitude distribution analysis," in *Proc. Interspeech*, 2008, pp. 2598–2601.

[21] T. Kinnunen, H. Delgado, N. Evans, K. A. Lee, V. Vestman, A. Nautsch, M. Todisco, X. Wang, M. Sahidullah, J. Yamagishi *et al.*, "Tandem Assessment of Spoofing Countermeasures and Automatic Speaker Verification: Fundamentals," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2195–2210, 2020.

[22] 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.