

CONTINUOUS MODELING OF THE DENOISING PROCESS FOR SPEECH ENHANCEMENT BASED ON DEEP LEARNING

Zilu Guo¹, Jun Du¹, Chin-Hui Lee²

¹University of Science and Technology of China, Hefei 230027, China

²Georgia Institute of Technology, Atlanta, GA. 30332-0250, USA

guozl@mail.ustc.edu.cn, jundu@ustc.edu.cn, chl@ece.gatech.edu

ABSTRACT

In this paper, we explore a continuous modeling approach for deep-learning-based speech enhancement, focusing on the denoising process. We use a state variable to indicate the denoising process. The starting state is noisy speech and the ending state is clean speech. The noise component in the state variable decreases with the change of the state index until the noise component is 0. During training, a UNet-like neural network learns to estimate every state variable sampled from the continuous denoising process. In testing, we introduce a controlling factor as an embedding, ranging from zero to one, to the neural network, allowing us to control the level of noise reduction. This approach enables controllable speech enhancement and is adaptable to various application scenarios. Experimental results indicate that preserving a small amount of noise in the clean target benefits speech enhancement, as evidenced by improvements in both objective speech measures and automatic speech recognition performance.

Index Terms— speech enhancement, speech denoising, controllable, controlling factor, UNet, denoising degree.

1. INTRODUCTION

Before the deep learning era, speech enhancement (SE) was performed in an unsupervised manner. Spectral subtraction [1], Weiner filter [2], minimum mean-square error (MMSE) [3], and optimally-modified log-spectral amplitude (OMLSA) [4] are prominent representatives of the conventional algorithms. To reduce the speech distortion and improve the perceptual quality, several noise estimation algorithms have been introduced to assist the denoising algorithms, such as minima controlled recursive averaging (MCRA) [5], and improved MCRA (IMCRA) [6]. In these approaches, the engineers can manipulate several hyper-parameters to balance speech quality and denoising intensity. For the deep-learning-based SE (DLSE), several mask-based SE methods have been proposed to mask the noise component in each time-frequency (T-F) bin, following the masking effect of the human ear [7]. A filter-like mask-based method is designed in [8]. Building upon this approach, researchers in [9, 10] have designed a low-complexity architecture for real-time SE. This method employs the clean signal mixed with a small portion of the noise as the training target to ensure the enhanced speech sounds comfortable. Recently, generative models have been booming in the SE, such as GANs [11], VAEs [12], Diffusion models [13, 14]. However, unlike conventional algorithms, the DLSE lacks hyper-parameters to control denoising intensity during the test stage. In other words, the denoising gain remains fixed and immutable once the training stage is completed. Preserving partial noise in the tar-

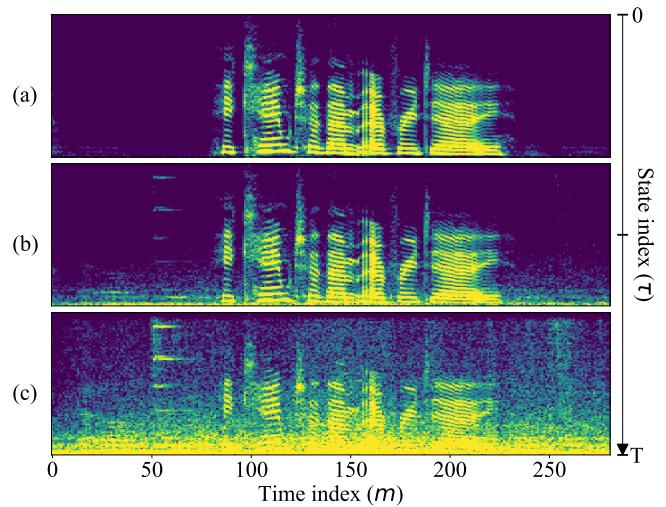


Fig. 1. Illustration of the process from clean to noisy speech. (a, b, c) are three spectrograms sampled from the process in different state indexes. (a) is the spectrogram of $s(0)$, i.e., the clean speech, (b) is the spectrogram of $s(\frac{T}{2})$, and (c) is the spectrogram of $s(T)$, i.e., the noisy speech.

get speech can reduce speech distortion, benefiting the automatic speech recognition (ASR) system [15, 16]. PL-based [16, 17] methods can alleviate this issue by providing the mid-outputs for different test scenarios. However, these algorithms offer only a limited set of mid-outputs, and adjusting the denoising intensity between them requires retraining for new applications. Taking Inspiration from conventional algorithms with denoising control parameters, human-control-based speech, or audio editing approaches [18, 19], we deploy a neural network (NN) to model the entire denoising process and introduce a controlling factor for UNet-based SE. This factor allows us to manipulate the denoising intensity during the enhancing stage. We incorporate the controlling factor as the input to the embedding in each UNet module and explore the impact on the performance through several embedding extractors.

2. THE PROPOSED METHOD

Given the additive noise $\mathbf{D}(n)$ and the clean speech $\mathbf{C}(n)$ the noisy speech $\mathbf{Y}(n)$ is denoted as

$$\mathbf{Y}(n) = \mathbf{C}(n) + \mathbf{D}(n) \quad (1)$$

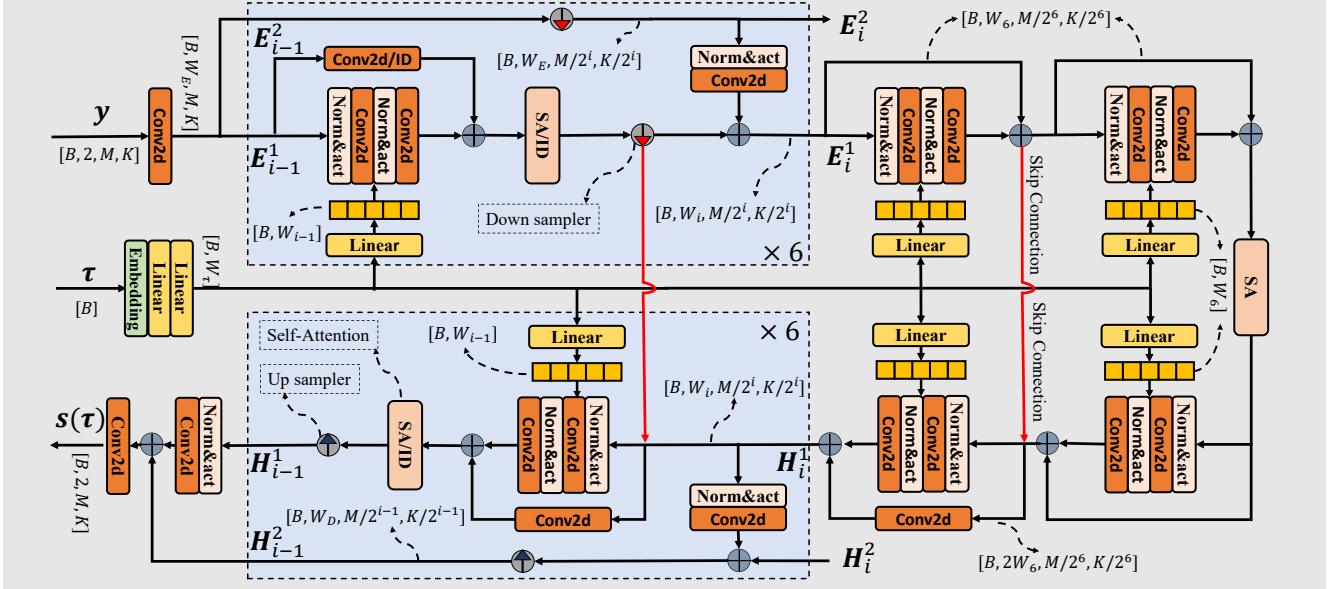


Fig. 2. Overview of the architecture of the controllable UNet. “ID” in “Conv2d/ID” and “SA/ID” denotes the Identity layer and the “/” signifies the implementation of either the former or the latter Identity layer. “SA” represents the channel-wise Self-Attention module.

where n is the time index in the time domain. Take the short-time Fourier transform (STFT) in both sides of Eq. 1, we get the representation of the noisy speech in the time-frequency (T-F) domain

$$\mathbf{y}(m, k) = \mathbf{c}(m, k) + \mathbf{d}(m, k) \quad (2)$$

where m, k represent the time and frequency indices, respectively, with the range $0 \leq m \leq M, 0 \leq k \leq K$, and M, K are the numbers of frames and discrete Fourier transform (DFT) bins respectively. The m, k will be omitted for simplicity.

2.1. The Signal Model

Unlike the perspective that the noisy speech in Eq. (1, 2) is obtained by directly adding the clean with the noise signal, we propose a different view. We consider a stochastic process between clean and noisy speech, where the clean part remains unchanged and the noise rises in a crescendo. In essence, the state variable starts as clean speech, becomes progressively noisier with the state index increment, and ultimately transforms into noisy speech. The state variable is defined as the linear combination of the clean speech \mathbf{c} and the noise \mathbf{d}

$$\mathbf{s}(\tau) = \mathbf{c} + \lambda(\tau) \cdot \mathbf{d} \quad (3)$$

$\mathbf{s}(\tau)$ is the state variable at the τ state index. $\tau \in [0, T] \in \mathbb{R}$ is the state index where T is the last state index. The function $\lambda(\tau) \in [0, 1] \in \mathbb{R}$ is the controlling factor that determines the process, and controls the changing speed from \mathbf{c} to \mathbf{y} , and manipulates the amplitude of the noise components in $\mathbf{s}(\tau)$. $\lambda(\tau)$ is a scalar function of τ and monotonic increasing when $\tau \in [0, T]$. $\lambda(0) = 0, \lambda(T) = 1$, therefore, $\mathbf{s}(0) = \mathbf{c}, \mathbf{s}(T) = \mathbf{y}$. In this paper, we set $\lambda(\tau)$

$$\lambda(\tau) = 1 - e^{-1.5\tau} \quad (4)$$

In practice, we set $T = 6$, which leads to $1 - \lambda(6) \approx 1.23 \cdot 10^{-4}$, making $\mathbf{s}(6) \approx \mathbf{y}$. In Fig. 1, we present three spectrograms of the state variables sampled from the process. The state index axis on the right side of Fig. 1 shows marks at $0, \frac{T}{2}$, and T corresponding to

three sampling indices. From the figure, it is evident that the magnitude of noise components gradually increases with the state index. The predefined process resembles the forward process in [13, 14] and also does not require any parameter to be learned. The denoising process from noisy speech to clean speech is defined within the set of state variables as τ decreases from T to 0. This process gradually removes the noise part from the noisy speech.

2.2. The Neural Network

The UNet-like architecture, known as *Noise Conditional Score Network* [20], is adapted for the SE task to learn the whole noise reduction process. The condition network is modified to incorporate the state index τ into every module of the neural network (NN) to estimate every state variable. Essentially, the controlling factor allows us to manipulate the degree of denoising. In this paper, we refer to the NN as Controllable UNet (CUNet). CUNet comprises a decoder and an encoder, each with two branches and a total of 7 blocks. The initial six blocks in both modules involve two up-sampling or down-sampling operations to extract high-level features or preserve different resolution features. The six modules of the encoder or decoder are connected sequentially, as indicated by the arrows in Fig. 2. The seventh block of the encoder is followed by two identical blocks that serve as the middle blocks. Additionally, a skip connection, denoted as the red line in Fig. 2, connects the seventh block of the encoder to the seventh block of the decoder. Group normalization and the Swish activation function are employed and labeled as “Norm&act” in Fig. 2. The upper branch of the encoder, denoted as \mathbf{E}_i^2 , attempts to downsample the low-level features and inject them into the stem branch (dubbed as \mathbf{E}_i^1). The lower branch of the decoder retains the high-level features and upsamples them. Consequently, the upsampled features are fused with the stem branch features, i.e., \mathbf{H}_i^1 . Additionally, we explore three different embedding extractors to enhance the performance.

The first embedding strategy we utilize is the sinusoidal posi-

tional embedding

$$\text{Emb}(\tau) = \begin{cases} \sin\left(\frac{\tau}{10000 W_{\text{emb}}^{2i}}\right), & i = 2l \\ \cos\left(\frac{\tau}{10000 W_{\text{emb}}^{2i}}\right), & i = 2l + 1 \end{cases} \quad (5)$$

where W_{emb} is the dimension of the embedding, $0 \leq i \leq W_{\text{emb}}$. The second strategy we utilize is the learnable Fourier embedding

$$\text{Emb}(\tau) = [\sin(2\pi\tau\theta), \cos(2\pi\tau\theta)] \quad (6)$$

where $\theta \in \mathbb{R}^{\frac{W_{\text{emb}}}{2}}$ represents the learnable parameter. In this approach, we do not use the embedding extractor. Instead, we expand the dimension of τ to create a matrix with the same dimension as the noisy spectrum, then concatenate this matrix with the noisy speech spectrum to form the input. The matrix is

$$\text{Emb}(\tau) = [\tau]_{M \times K} \quad (7)$$

The real and imaginary parts of the complex spectrum \mathbf{y} are stacked as two channels of the input. Both parts are scaled before being fed into the NN to mimic the non-linearity of auditory perception of loudness. Taking the spectrum of noisy speech \mathbf{y} as an example, the element-wise scale function ϕ is

$$\phi(\mathbf{y}) = a|\mathbf{y}|^b e^{\angle \mathbf{y}} = a \left[\frac{\mathbf{y}_r}{|\mathbf{y}|^{1-b}} + j \frac{\mathbf{y}_i}{|\mathbf{y}|^{1-b}} \right] \quad (8)$$

where j denotes the imaginary unit, $[\cdot]_r$ and $[\cdot]_i$ are the real and imaginary parts of the complex spectrum respectively, and $|\cdot|$ calculates the norm of a complex-valued number. $\angle \cdot$ computes the angle of a complex-valued number. a and b are hyper-parameters. In this paper, we set $a = 0.15$ and $b = 0.5$.

The inverse function is defined when we reconstruct the complex spectrum

$$\mathbf{y} = \phi^{-1}(\phi(\mathbf{y})) = \frac{|\phi(\mathbf{y})|^{-b}}{a} e^{\angle \phi(\mathbf{y})} \quad (9)$$

2.3. The Cost Function

Commonly, a well-designed neural network (NN) ψ is utilized to predict the clean speech \mathbf{c} , which takes as the input \mathbf{y} . Under the minimum mean square error (MMSE) criterion, the cost function of the SE task is

$$\mathcal{L} = \mathbb{E}_{(\mathbf{c}, \mathbf{y})} \|\phi(\mathbf{c}), \phi[\psi(\mathbf{y})]\|_2^2 \quad (10)$$

\mathbb{E} is the expectation, $\|\cdot\|_2^2$ is the square of the Fibonacci norm of a matrix, and $\psi_\theta(\mathbf{y})$ is the output of the NN. We treat the NN differently by training it to master the whole process defined in Eq. 3. As a result, the NN can output the estimate $\mathbf{s}(\tau) = \psi(\mathbf{y}, \tau)$ for any given state index τ and the noisy speech \mathbf{y} . In practice, the $\mathbf{s}(\tau)$ close to the noisy speech is not needed for most scenarios. Therefore, NN does not necessarily have to learn these values. Without loss of generality, let us assume that $1 < T$. In this case, the NN attempts to estimate all state variables when $\tau \in [0, 1]$. Here, $\psi(\mathbf{y}, 0)$ corresponds to the maximum denoising and $\psi(\mathbf{y}, 1)$ corresponds to the least denoising. The cost function is

$$\mathcal{L} = \int_{\tau=0}^1 \{\mathbb{E}_{(\mathbf{c}, \mathbf{y})} \|\phi[\mathbf{s}(\tau)], \phi[\psi(\mathbf{y}, \tau)]\|_2^2\} d\tau \quad (11)$$

However, it is not practicable to calculate the integral in Eq. 11. Because it is extremely time-consuming to compute the loss once until the NN traverses all τ . We transmute the cost function into

$$\mathcal{L} = \mathbb{E}_{(\mathbf{c}, \mathbf{y}), \tau \sim \mathcal{U}(0, 1)} \|\phi[\mathbf{s}(\tau)], \phi[\psi(\mathbf{y}, \tau)]\|_2^2 \quad (12)$$

Table 1. The performance comparison of the proposed approach and other SOTA approaches on the VBD simulation test set.

Model	PESQ \uparrow	CSIG \uparrow	CBAK \uparrow	COVL \uparrow
noisy	1.97	3.35	2.44	2.64
UNet	2.88	3.87	3.49	3.39
CUNet-SP ($\tau = 0$)	2.90	3.81	3.50	3.28
CUNet-SP ($\tau = 0.12$)	2.92	4.10	2.84	3.51
CUNet-CAT ($\tau = 0$)	2.85	3.75	3.38	3.28
CUNet-CAT ($\tau = 0.16$)	3.00	4.15	2.79	3.57
CUNet-LF ($\tau = 0$)	2.87	3.67	3.45	3.26
CUNet-LF ($\tau = 0.12$)	3.05	4.23	2.90	3.62
SGMSE+ [13]	2.93	4.12	3.37	3.51
VPIIDM [14]	3.13	4.63	3.41	3.94

where $\mathcal{U}(0, 1)$ is the uniform distribution from 0 to 1. In practice, we use the approximate form of this expectation

$$\mathcal{L} = \frac{\sum_{m=0, k=0, b=1}^{M-1, K-1, B} \|\phi[\mathbf{s}_b(m, k, \tau_b)] - \phi[\psi(\mathbf{y}_b(m, k), \tau_b)]\|^2}{K \cdot M \cdot B} \quad (13)$$

where B is the number of batch sizes, \mathbf{s}_b , \mathbf{y}_b , and τ_b are the b -th state variable, noisy speech, and state index in a mini-batch, respectively.

3. EXPERIMENTS AND ANALYSIS

In our experimental setup, we use a convolutional neural network (CNN)-based UNet architecture with kernel sizes consistent with previous studies in [13, 14, 20]. The channel sizes of the convolution layers are as follows: $\{W_i\}_0^6 = \{128\}_{\times 2}, \{256\}_{\times 5}$, W_E is set to 128, W_D is set to 4, and both W_τ and W_{emb} are 512. We apply a Hanning window with a window length of 510, a hop length of 128, and a sample rate of 16 kHz. All input signals are padded or cut to achieve a frame length of 256, resulting in the M, K being set to 256. The batch size is chosen as 32. For our investigation, we empirically select 25 sampling indices uniformly in $[0, 1]$ to observe state variables. CUNet-SP utilizes the embedding specified in Eq. 5, CUNet-LF is the model with the embedding from Eq. 6, and CUNet-CAT uses the embedding described in Eq. 7. Additionally, we implement the NN to estimate only the clean spectrum, resembling the conventional deep-learning-based method, denoted as UNet. In this configuration, the parameter τ remains fixed to a constant for all input noisy speech. In our experiments, we use the VoiceBank+DEMAND (VBD) [21] dataset. From the test dataset, we randomly selected 25 out of 824 clips to form the validation dataset. During the validation stage, we only evaluate the performance of the estimated $\mathbf{s}(0)$ corresponding to the clean signal. This evaluation is used to select the best checkpoint based on perceptual evaluation of speech quality (PESQ) [22]. We evaluate our model using three objective mean opinion scores (MOS), signal distortion (CSIG), background intrusiveness (CBAK), and overall quality (COVL) as defined in [23]. These scores help us select the optimal embedding extractor and compare the performance of the proposed method to other state-of-the-art methods. Interestingly, the PESQ, CSIG, and COVL exhibit similar MOS curves across different state indices. Therefore, only the CBAK and COVL curves are drawn to investigate the performance of each state variable in the denoising process (Figs. 3 and 4). Sinusoidal embedding fails to provide sufficient state information for the UNet to reduce noise. Surprisingly, the best COVL MOS is not achieved when the NN attempts to remove all noise at $\tau = 0$, suggesting that retaining a

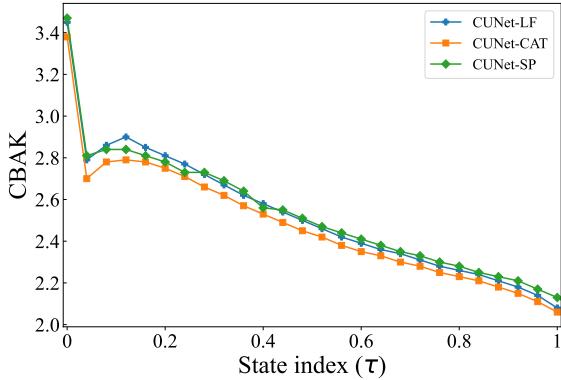


Fig. 3. The CBAK MOS comparison of the proposed approach with different state indices for 3 types of embeddings.

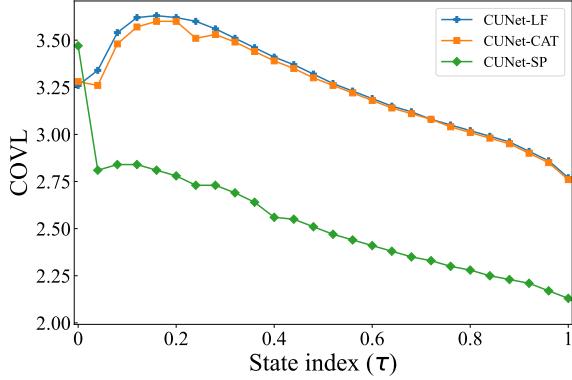


Fig. 4. The COVL MOS comparison of the proposed approach with different state indices for 3 types of embeddings..

small amount of noise is beneficial for overall speech quality. However, this strategy is not effective for the CBAK, as preserving any noise in the enhanced speech will decrease the performance of the background-noise MOS performance. The UNet-LF outperforms CUNet-CAT slightly, making it our optimal model for the subsequent experiments. From Tab. 1, when maximizing the noise reduction, the three embeddings achieve similar performance to the conventional UNet. We also compare our model to two SOTA methods that employ nearly identical NNs for SE tasks. While our model does not surpass VPIDM, it achieves competitive performance with only one sampling step compared to VPIDM's 25.

We also conduct experiments on the CHiME 4 dataset [24] to validate the ASR performance of the proposed method, i.e., the word error rate (WER). Our model is trained on the simulated dataset and tested on the dataset recorded in the four real noise environments, namely, cafe (CAF), street junction (STR), public transport (BUS), and pedestrian area (PED). Training settings mirror those in the CUNet-LF. The training dataset follows the methodology described in [17]. For validation, we randomly selected 25 segments from the simulated development dataset. To evaluate the ASR performance, the off-the-shelf pre-trained backend from [24] is adopted without joint training. From Fig. 5, the best WER in the BUS noise

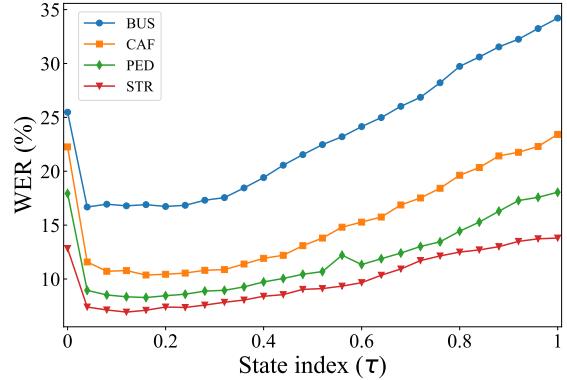


Fig. 5. The WER comparison of the proposed approach under four noisy conditions on the real test set of CHiME-4.

Table 2. The overall comparison of our proposed approach and other SOTA approaches.

Model	The WER (%) on the real test set of CHiME-4					The simulated	
	BUS ↓	CAF ↓	PED ↓	STR ↓	Avg. ↓	PESQ ↑	STOI ↑
Noisy	36.55	24.73	19.92	14.16	23.84	1.98	0.811
UNet	25.48	22.26	19.74	12.83	19.63	2.59	0.883
PL-ANSE [17]	27.99	20.56	15.81	9.92	18.57	-	-
ConvPL [25]	18.99	13.30	11.17	8.72	13.05	2.67	0.908
CUNet-LF ($\tau = 0.04$)	16.68	11.55	8.95	7.40	11.15	2.85	0.919
CUNet-LF ($\tau = 0.16$)	16.90	10.03	8.28	7.10	10.66	2.64	0.908

environment is achieved at $\tau = 0.04$, that in the STR environment at $\tau = 0.12$ and those in CAF and PED environments at $\tau = 0.16$. This corresponds to an approximate improvement of 25 dB, 16 dB, and 13 dB in the signal-to-noise ratio (SNR) respectively. Comparing these results to the perceptual quality in Fig. 4 and 3, the optimal performance is always achieved when preserving some noise components. However, the WER curves exhibit a longer flat area than the two MOS curves, suggesting that the ASR is more robust for noise to some extent. Additionally, we have observed a significant WER gap between $\tau = 0$ (SNR improvement $\rightarrow +\infty$) and $\tau = 0.04$ (SNR improvement ≈ 25 dB). The gap can be attributed to inevitable speech distortion, particularly artifacts introduced when the algorithm enhances the noisy speech. The more noise reduction, the more artifacts. We also conduct experiments to assess the perceptual measures, yielding results similar to those in Fig. 4 and 3, although we do not display the results due to space constraints. From Tab. 2, we can see that the optimal objective metrics are obtained when $\tau = 0.04$ (SNR improvement ≈ 25 dB), interestingly, the best WER is observed at $\tau = 0.16$ (SNR improvement ≈ 13 dB), highlighting that the perceptual quality demands more noise reduction than the ASR task.

4. CONCLUSIONS

In this paper, we treat the denoising process as a continuum and deploy an NN to estimate every state variable sampled from the continuum. The results show that retaining a small amount of noise in the enhanced speech leads to performance improvement in both speech perceptual quality and ASR tasks, and that the ASR task is more resilient to noise than the perceptual quality. Moreover, one can tailor the intensity of noise reduction to achieve the desired outcome depending on the application scenario based on the controlling factor.

5. REFERENCES

- [1] Steven Boll, “Suppression of acoustic noise in speech using spectral subtraction,” *IEEE Transactions on acoustics, speech, and signal processing*, vol. 27, no. 2, pp. 113–120, 1979.
- [2] Jingdong Chen and Yiteng Huang, “New insights into the noise reduction wiener filter,” *IEEE Transactions on audio, speech, and language processing*, vol. 14, no. 4, pp. 1218–1234, 2006.
- [3] Y. Ephraim and D. Malah, “Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 6, pp. 1109–1121, 1984.
- [4] Israel Cohen and Baruch Berdugo, “Speech enhancement for non-stationary noise environments,” *Signal processing*, vol. 81, no. 11, pp. 2403–2418, 2001.
- [5] Israel Cohen and Baruch Berdugo, “Noise estimation by minima controlled recursive averaging for robust speech enhancement,” *IEEE signal processing letters*, vol. 9, no. 1, pp. 12–15, 2002.
- [6] Israel Cohen, “Noise spectrum estimation in adverse environments: Improved minima controlled recursive averaging,” *IEEE Transactions on speech and audio processing*, vol. 11, no. 5, pp. 466–475, 2003.
- [7] Yuxuan Wang, Arun Narayanan, and DeLiang Wang, “On training targets for supervised speech separation,” *IEEE/ACM transactions on audio, speech, and language processing*, vol. 22, no. 12, pp. 1849–1858, 2014.
- [8] Wolfgang Mack and Emanuël AP Habets, “Deep filtering: Signal extraction and reconstruction using complex time-frequency filters,” *IEEE Signal Processing Letters*, vol. 27, pp. 61–65, 2019.
- [9] Hendrik Schröter, Alberto N. Escalante-B., Tobias Rosenkranz, and Andreas Maier, “DeepFilterNet: A low complexity speech enhancement framework for full-band audio based on deep filtering,” in *ICASSP 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022.
- [10] Hendrik Schröter, Alberto N. Escalante-B., Tobias Rosenkranz, and Andreas Maier, “DeepFilterNet2: Towards real-time speech enhancement on embedded devices for full-band audio,” in *17th International Workshop on Acoustic Signal Enhancement (IWAENC 2022)*, 2022.
- [11] Szu-Wei Fu, Cheng Yu, Tsun-An Hsieh, Peter Plantinga, Mirco Ravanelli, Xugang Lu, and Yu Tsao, “MetricGAN+: An Improved Version of MetricGAN for Speech Enhancement,” in *Proc. Interspeech 2021*, 2021, pp. 201–205.
- [12] Huajian Fang, Guillaume Carbajal, Stefan Wermter, and Timo Gerkmann, “Variational autoencoder for speech enhancement with a noise-aware encoder,” in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 676–680.
- [13] Simon Welker, Julius Richter, and Timo Gerkmann, “Speech enhancement with score-based generative models in the complex STFT domain,” in *Proc. Interspeech 2022*, 2022, pp. 2928–2932.
- [14] Zilu Guo, Jun Du, Chin-Hui Lee, Yu Gao, and Wenbin Zhang, “Variance-Preserving-Based Interpolation Diffusion Models for Speech Enhancement,” in *Proc. INTERSPEECH 2023*, 2023, pp. 1065–1069.
- [15] Kazuma Iwamoto, Tsubasa Ochiai, Marc Delcroix, Rintaro Ikeshita, Hiroshi Sato, Shoko Araki, and Shigeru Katagiri, “How bad are artifacts?: Analyzing the impact of speech enhancement errors on ASR,” in *Proc. Interspeech 2022*, Hanseok Ko and John H. L. Hansen, Eds. 2022, pp. 5418–5422, ISCA.
- [16] Yan-Hui Tu, Jun Du, Tian Gao, and Chin-Hui Lee, “A multi-target snr-progressive learning approach to regression based speech enhancement,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1608–1619, 2020.
- [17] Zhaoxu Nian, Jun Du, Yu Ting Yeung, and Renyu Wang, “A time domain progressive learning approach with snr constriction for single-channel speech enhancement and recognition,” in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6277–6281.
- [18] Haixin Zhao, “A GAN Speech Inpainting Model for Audio Editing Software,” in *Proc. INTERSPEECH 2023*, 2023, pp. 5127–5131.
- [19] Rongjie Huang, Jiawei Huang, Dongchao Yang, Yi Ren, Luping Liu, Mingze Li, Zhenhui Ye, Jinglin Liu, Xiang Yin, and Zhou Zhao, “Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models,” in *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*. 2023, vol. 202 of *Proceedings of Machine Learning Research*, pp. 13916–13932, PMLR.
- [20] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole, “Score-based generative modeling through stochastic differential equations,” *arXiv preprint arXiv:2011.13456*, 2020.
- [21] Cassia Valentini-Botinhao, Xin Wang, Shinji Takaki, and Junichi Yamagishi, “Investigating rnn-based speech enhancement methods for noise-robust text-to-speech.,” in *SSW*, 2016, pp. 146–152.
- [22] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra, “Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs,” in *2001 IEEE international conference on acoustics, speech, and signal processing. Proceedings (Cat. No. 01CH37221)*. IEEE, 2001, vol. 2, pp. 749–752.
- [23] Yi Hu and Philipos C Loizou, “Evaluation of objective quality measures for speech enhancement,” *IEEE Transactions on audio, speech, and language processing*, vol. 16, no. 1, pp. 229–238, 2007.
- [24] Emmanuel Vincent, Shinji Watanabe, Aditya Arie Nugraha, Jon Barker, and Ricard Marxer, “An analysis of environment, microphone and data simulation mismatches in robust speech recognition,” *Comput. Speech Lang.*, vol. 46, pp. 535–557, 2017.
- [25] Zhaoxu Nian, Jun Du, Yu Ting Yeung, and Renyu Wang, “A time domain progressive learning approach with SNR constriction for single-channel speech enhancement and recognition,” in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022, Virtual and Singapore, 23-27 May 2022*. 2022, pp. 6277–6281, IEEE.