Proceedings of CECNet 2022
A.J. Tallón-Ballesteros (Ed.)

doi:10.3233/FAIA220555

Tiny Deep Convolution Recurrent Network for Online Speech Enhancement with Various Noise Types

Qiuwei DENG[abc,1], Di WANG[c], Tianxiang LUAN[c] and Bin HAO[c]
[a] *College of Computer Science, Chongqing University, Chongqing, China*
[b] *State Key Laboratory of Digital Household Appliances, Qingdao, China*
[c] *Haier Smart-Home Intelligent Technology co. LTD, Qingdao, China*

Abstract. Nowadays, voice interaction is increasingly applied to smart home appliances. There are various types of noises in our real lives, which requires speech enhancement technology to deal with multiple noisy speech scenarios and to process them in real-time. Traditional technologies of speech noise reduction require estimating the noise power spectrum first, then estimating the spectrogram gain value of noise reduction, such as minima controlled recursive averaging (MCRA), which can only deal with stationary environmental noises but cannot estimate noises with serious fluctuations of the power spectrum within quite limited durations. A highly complicated deep-learning model can estimate the power spectrum of various types of noise, but it cannot meet the requirement of real-time processing due to the large number of parameters of these general models. In this paper, we proposed a method combining deep-learning technologies with traditional signal processing techniques to estimate the power spectrum of various types of noises by designing a new model with fewer parameters, tiny deep convolutional recurrent network (TDCRN), and computing the speech gain value with the power spectrum. The result of our experiment indicates that, compared with the traditional technology and complicated deep-learning model, the proposed method, with only 0.29M parameters, increases the PESQ by more than 0.6, the STOI by more than 0.2 and the wake-up rate by more than 6%.

Keywords. deep learning, real-time speech enhancement, noise power spectrum estimation, convolutional encoder-decoder

1. Introduction

The speech processing technology is increasingly applied in hearing aids, automatic speech recognition, and audio/video calls. Due to the interference of noise in actual scenarios, the quality of signals received by microphones is degraded, which seriously affects the subsequent use. Speech enhancement technologies can be used to improve the acoustic quality and speech intelligibility of speech signals whose quality has been degraded due to additive noise.

Traditional speech enhancement technologies include spectral subtraction [1], Wiener filter [2] and estimation based on noise spectrum [3], among which the more

[1]Corresponding Author: Qiuwei Deng; E-mails: dengqiuwei@haier.com

well-developed and widely-used is the estimation based on noise spectrum. Cohen assumes that the Fourier transform of clean speech and noise satisfies a Gaussian distribution, using the optimally-modified log-spectral amplitude (OM-LSA) with minimum Bayesian estimation to solve for the optimal gain, during which the noise power spectrum is estimated with improved minima controlled recursive averaging (IMCRA) [4].

Recently, deep learning has made great achievements in the application of speech enhancement. The mainstream approach is to carry out speech enhancement in the TF domain. Xu[5] proposed the method of using DNN to directly learn the spectral mapping between the noise speech and the clean speech so as to obtain a clean speech spectrum, the training of which can be divided into two parts: the pre-training and the refined adjustment based on MMSE. One method is to conduct speech enhancement by masking[6][7][8]. We assume that both noise signal and speech signal exist in the noisy speech signal, and the speech signal can be left after masking out the noise signal. Currently, there are two masking methods: ideal binary mask (IBM) and ideal ratio mask (IRM). Recently, complex networks have also become popular, which can exploit phase information and have higher upper-performance limits theoretically[9][10][11] compared to real networks, but also require a large number of parameters.

The traditional technology cannot eliminate the noise that has a greater fluctuation in the power spectrum and short duration. Using deep learning to estimate the IRM, the enhanced signal obtained has speech distortion and aberration. If complex networks are used, a large number of parameters are required, which cannot meet the requirements of real-time processing. In this paper, we proposed a new method of speech enhancement, which is inspired by the method of speech processing based on DNN-mask[6][7][8]. The noise-reduction method we proposed introduces deep learning into the estimation of noise spectrum, using networks to output the mask value of noise components in the noisy speech signal, and making further estimation on priori SNR and posteriori SNR of the speech as well as the missing probability of priori speech, so as to obtain speech presence probability. As for the model result, we adjusted the CRN[12] structure and incorporated a convolutional encoder-decoder and long short-term memory. Compared with the LSTM model, CRN has better performance in objective speech intelligibility and quality.

The arrangements of this paper are as follows. In Section 2, we formulate the problem of speech noise reduction. In Section 3, we review the noise reduction method based on OM-LSA and IMCRA, and also describe its features and limitations. Section 4 introduces the noise-reduction method that we proposed. Section 5 provides the experimental results, and demonstrates the improvement of ASR by the noise-reduction method that we proposed in a real noise environment. Last but not least, we put our conclusions in Section 6.

2. Problem Formulation

In this section, we describe the problem of speech noise reduction in the TF domain. $d(t)$ and $n(t)$ stand for speech signal and additive noise signal respectively, and signal received by microphone $y(t)$ can be expressed as $y(t) = d(t) + n(t)$. With a short-time Fourier transform, it can be expressed as $Y(t,k) = D(t,k) + N(t,k)$, among which t and k stand for frame index and frequency bin index respectively.

As long as an accurate gain value $G(t,k)$ can be obtained, an estimated clean signal $\widehat{D}(t,k)$ can be obtained as well.

$$\widehat{D}(t,k) = G(t,k)Y(t,k) \tag{1}$$

2.1. Spectral Gain

The computing criterion of OM-LSA is to minimize the error in the optimally-modified log-spectral amplitude of actual clean speech and estimated clean speech $E\left\{\left[log|D(t,k)| - log\left|\widehat{D}(t,k)\right|\right]^2\right\}$. Assuming the statistical independence of spectral components[13], the log-spectral amplitude of clean speech is

$$\left|\widehat{D}(t,k)\right| = exp\{E[log|D(t,k)|] \mid Y(t,k)\}. \tag{2}$$

A binary hypothesis model is set, and $H_0(t,k)$ and $H_1(t,k)$ stand for the non-presence and presence of the speech respectively.

$$H_0(t,k): Y(t,k) = N(t,k)\,,$$

$$H_1(t,k): Y(t,k) = D(t,k) + N(t,k). \tag{3}$$

The spectrogram gain value of OM-LSA[3] can be calculated as:

$$G(t,k) = \{G_{H1}(t,k)\}^{p(t,k)} G_{min}^{1-p(t,k)},$$

$$G_{H1}(t,k) = \frac{\xi(t,k)}{1+\xi(t,k)} exp\left(\frac{1}{2}\int_{\upsilon(t,k)}^{\infty} \frac{e^{-t}}{t} dt\right). \tag{4}$$

In the above equation, $\xi(t,k) \triangleq \frac{\lambda_d(t,k)}{\lambda_n(t,k)}$, $\gamma(t,k) \triangleq \frac{|Y(t,k)|^2}{\lambda_n(t,k)}$, $\upsilon(t,k) \triangleq \frac{\gamma(t,k)\xi(t,k)}{1+\xi(t,k)}$. $\xi(t,k)$ and $\gamma(t,k)$ stand for priori SNR and posteriori SNR respectively. $\lambda_d(t,k)$ and $\lambda_n(t,k)$ stand for the power spectrum of clean signals and noise signals respectively. $p(t,k)$ and $q(t,k)$ stand for the priori probability of speech presence and non-presence respectively.

2.2. Speech Presence Probability

Assuming that the short-time Fourier transform coefficients of speech and noise conform to the complex Gaussian distribution and are incoherent, according to Bayes theorem, the presence probability of conditional speech is

$$p\left(H_1(t,k)\middle|Y(t,k)\right) = \frac{1}{1+\frac{q(t,k)}{1-q(t,k)}\times[1+\xi(t,k)]\times exp\{-\upsilon(t,k)\}}. \tag{5}$$

Among which priori SNR can be estimated based on historical smoothing information,

$$\hat{\xi}(t,k) = \alpha {G_{H1}}^2(t-1,k)\gamma(t-1,k) + (1-\alpha)max\{\gamma(t,k)-1,0\}. \quad (6)$$

The first term on the right of Eq. (6) can be interpreted as the estimation of priori SNR on the last frame, the second term can be interpreted as the estimation of priori SNR on the current frame, and the final estimation of priori SNR can be obtained by smoothing the above two parts.

3. Conventional Method

To solve for the final spectrogram gain value $G(t,k)$, it is necessary to accurately solve for the spectrogram gain value when the speech is present $G_{H1}(t,k)$, as well as the presence probability of conditional speech $p\big(H_1(t,k)\big|Y(t,k)\big)$, during which the most vital unknown parameter is noise power spectrum $\lambda_n(t,k)$.

3.1. Formulation

The commonly used method for noise spectrum estimation, MCRA[3], combines recursive averaging and minimum value tracking to roughly estimate the speech presence probability based on minimum value tracking, eliminate the frequency points with high speech probability, filter out the noisy segments, and then update the noise spectrum only in the noisy segments. Cohen proposed to improve MCRA[4] by determining the frequency points with higher speech probability that need to be eliminated by two iterations, expanding the historical window of minimum tracking, and empirically compensating for the final estimated noise spectrum.

$$\hat{\lambda}_n(t,k) = \tilde{\alpha}_n(t,k)\hat{\lambda}_n(t-1,k) + \big(1-\tilde{\alpha}_n(t,k)\big)|Y(t,k)|^2 \quad (7)$$

The noise spectrum estimation in Eq. (13) is all obtained by smoothing the historical power spectrum by adjusting the smoothing parameter based on the speech presence probability, and $\tilde{\alpha}_n$ is the smoothing factor obtained based on the speech presence probability.

3.2. Limitations of Conventional Method

The MCRA method of estimating the noise spectrum has the following limitations: 1. The problem of convergence; 2. It can only be used for estimating environmental noises which are rather smooth.

$$\tilde{\alpha}_n(t,k) = \alpha_n + (1-\alpha_n)p_n(t,k) \quad (8)$$

In the above equation, α_n is a smoothing coefficient, and also a constant.

When the noise environment changes, the method of recursive averaging requires a certain number of frames to reach convergence. In Eqs. (6), (7), and (8), smoothing methods are used in estimating priori SNR, noise spectrum, and smoothing coefficient of the noise spectrum. Moreover, in the method of IMCRA, when tracking the minimum value of the power spectrum by two iterations, the minimum power spectrum saved during the smoothing process is calculated in the first iteration. In the second

iteration, the amount of windows D is divided into U shares, each with V sampling points, which also calculates the minimum power spectrum of the U shares. While this method improves the accuracy of noise spectrum estimation and eliminates those relatively strong speech components, it also increases the convergence time.

The accuracy of noise spectrum estimation depends on the estimation of the smoothing coefficient $\tilde{\alpha}_n$. The most ideal scenario is the frequency band with a high noise component and a small value of $\tilde{\alpha}_n(t,k)$. The first term on the right of Eq. (7) has a small proportion of historical values, and the second term has a large proportion of current noise spectral components. Frequency bands with higher speech components have higher values of $\tilde{\alpha}_n(t,k)$, and the accuracy of $\tilde{\alpha}_n(t,k)$ value depends on $p_n(t,k)$. In the method of IMCRA, Eq. (5) is used to estimate $p_n(t,k)$, and the minimum power spectrum S_{min} estimated with secondary iterations is regarded as λ_n. S_{min} calculates the minimum value of the power spectrum after time and frequency smoothing, and the computed $p_n(t,k)$ can only distinguish the frequency points with greater fluctuations in the power spectrum. We consider these points to be the frequency points with a higher probability of speech. The noise power spectrum $\lambda_n(t,k)$ obtained by Eq. (7) is rather smooth in the time dimension. Therefore, the method of OM-LSA & MCRA can only suppress the environmental noise that is relatively smooth, and cannot eliminate the noise that has a greater fluctuation in the power spectrum and short duration, such as keyboard tapping, knocking on tables and chairs, frying in the kitchen, etc.

4. Proposed Method: CRN-based Noise Spectrum Estimation

4.1. The Proposed Network Architecture

In order to solve the above problems, we proposed a new computing method to estimate the required noise power spectrum λ_n, instead of using the methods of recursive averaging and minimum tracking. Specifically, we proposed to use deep learning to estimate λ_n and apply it to estimate the priori SNR ξ, so as to compute the spectrogram gain value of noise reduction.

The CRN network structure proposed by Tan[12] uses CED (convolutional encoder-decoder) and long short-term memory (LSTM) in the model structure, which can be used for the real-time processing of speech signals. Choi[11] proposed a small UNET network that implements CED with one-dimensional convolution to reduce the number of trainable parameters.

In this paper, we proposed the tiny deep convolution recurrent network (TDCRN) based on previous network structures, using the loss function of mean squared error (MSE) to perform network optimization. This network model effectively combines the advantages of UNET and CRN methods, using LSTM to model the temporal dependencies. The encoder and decoder are implemented by one-dimensional convolution in the time domain, which can effectively reduce the number of trainable parameters and computing overhead.

A. Model input and output

The division of Bark band utilizes the perceptual characteristics of human ears to meticulously detail the low-frequency components of the signal. Setting fewer Bark bands can reduce the amount of computation and memory. The scale of Bark domain

nonlinearly maps frequencies to the perception domain of human ears. b stands for critical frequency band, and its relation with frequency f is,

$$b = 13 \times \arctan(0.76 \times f) + 3.5 \times \arctan\left(\frac{f}{7.5}\right)^2. \quad (9)$$

The amplitude spectrum of noisy speech signal is mapped to the Bark domain, and used as model input. Model output is the ratio of amplitude spectrum of clean signal to that of noisy signal in the Bark domain. $\mathcal{G}(\cdot)$ stands for model output, and $\mathcal{F}(nn)$ denotes the value mapped from the bark domain to the frequency domain. The estimated noise amplitude spectrum is,

$$\hat{\lambda}_n(t,k) = \mathcal{F}(nn)|Y(t,k)|^2. \quad (10)$$

B. Model structure

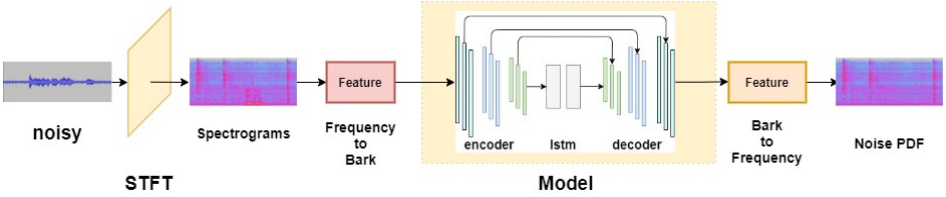

Figure 1. Flow Chart of TDCRN Estimation Noise Spectrum

The flow chart of our proposed TDCRN model for estimating the noise spectrum is shown in Figure 1. In this paper, we set up 3 layers of convolutional networks in encoder and decoder respectively, as shown in Figure 1. The output of each layer of encoder is used as part of the input of decoder layer with the method of res-net. Compared with [9][12], we use one-dimensional convolution instead of two-dimensional convolution. The convolution direction of [11] is the frequency direction of a frame, while the convolution direction that we proposed is the frame direction. What is shown in Figure 2 is the computing method of a convolution kernel in one-dimensional convolution. We set in_channels, a one-dimensional convolution parameter, as the feature number of input, and out_channels as the feature vector of output. This convolutional approach can extend the perceptual field of the network, and also count the influence among frequencies.

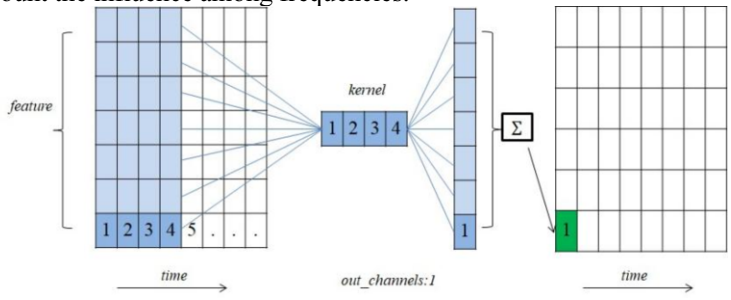

Figure 2. One-dimensional Convolution Diagram

4.2. Advantage of Proposed Method

Compared with the traditional method of estimating the noise spectrum, a major advantage of using the deep learning method to estimate the noise power spectrum λ_n is that it can estimate the change of noise in time and greatly reduce the convergence time of noise reduction. Besides, a reasonably trained model can learn the power spectrum of a wide variety of noises, and noise reduction is no longer limited to environmental noise that is rather smooth.

If the model output is used as the final output, a large number of parameters are often required to reduce speech distortion and aberration, which cannot meet the demand of real-time processing. In contrast, the method of combining TDCRN with OM-LSA only requires TDCRN to estimate the noise spectrum, without the need to consider the complex information of speech spectrum structure, so that it is advisable to design the model structure with fewer parameters.

Rnnoise [14] is a hybrid method based on the improved traditional methods. In order to ensure the real-time performance of the algorithm, the network output is the 22 band in the bark domain. Although the computing resource of this method is low and the noise reduction ability is increased, the network output dimension is too small, some details are missing, and the speech distortion is relatively serious.

5. Experiment

5.1. Datasets

In the experiment, there are 62,810 noise data from Interspeech2021 DNS Challenge[15] dataset in the noise dataset, as well as 6.2 hours of free-sound noise and 42.6 hours of music in MUSAN[16] dataset. Besides, we recorded the 100-hour noise from an actual kitchen scenario with a microphone, including noises from frying and chopping food, hood, high-speed blender, etc.

In the clean speech dataset, there are 178 hours of human voice recorded by AISHELL-1[17]. We used the method of IMAGE to generate 1,000 rir with the clean data from Interspeech2021 DNS Challenge, and get the human voice date through convolution. Moreover, we also recorded 100 hours of human voice in an anechoic chamber with a microphone.

We generated 100,000 mixed data as the training dataset, each with 4s, which were 111 hours in total. We set SNR as {-5,0,5,10}dB. We generated 15,000 as the test dataset, and 7,500 as the validation dataset, in which the percentage of real data is increased.

5.2. Training Setup and Baselines

The data are all with 16000Hz sampling rate, 32ms of STFT window length, 16ms of frameshift, and 512 of FFT length. The model was trained by Pytorch, optimizer Adam, with 0.001 initialized learning rate and 64 batch size.

The details of parameters of TDCRN model are shown in Table 1, with feature dimension of model output as 128, and the Hyper-parameters in Encoder and Decoder denote [in_channels, out_channels] and [kernel_size, stride, padding] respectively. In

Encoder, we chose to discard the first layer of data due to the output of T+1 per layer of zeros_padding. Similarly, in Decoder, we conducted zero-padding at the first layer due to the output of T-1 per layer of zeros_padding. The number of parameters for the whole model is 0.29M.

Table 1. Proposed TDCRN Structure. T stands for frames, and B stands for batch size.

	Layer name	***Input size***	***Hyper-parameters***	***Output size***
	Stft	[B, time]		[B, F,T]
	Frequency to Bark	[B, F,T]		[B, 128,T]
Encoder	Conv1d_1	[B, 128,T]	[128,96],[4,1,2]	[B, 96,T]
	Conv1d_2	[B, 96,T]	[96,64],[4,1,2]	[B, 64,T]
	Conv1d_3	[B, 64,T]	[64,48],[4,1.2]	[B, 48,T]
RNN	Lstm_1	[B, T,48]	[48,64]	[B, T,64]
	Lstm_2	[B, T,64]	[64, 48]	[B, T,48]
Decoder	Conv1dTranspose_1	[B, 48,T]+ [B, 48,T]	[48+48,64],[4,1,2]	[B, 64,T]
	Conv1dTranspose_2	[B, 64,T]+ [B, 64,T]	[64+64,96],[4,1,2]	[B, 96,T]
	Conv1dTranspose_3	[B, 96,T]+ [B, 96,T]	[96+96,128],[4,1.2]	[B, 128,T]

We respectively chose IMCRA and LSTM models as the baselines.

LSTM: A semi-causal model consists of two LSTM layers, each containing 128 units. The output layer is a 128-unit fully connected layer with sigmoid activation function. The number of model parameters is 0.27M.

IMCRA: The coefficients are set according to[4].

At last, we compared the algorithm performance of IMCRA&OMLSA, LSTM out, TDCRN out, and TDCRN & OMLSA, among which LSTM out and TDCRN out models were trained when the target was with clean signals.

5.3. Results and Discussions

In this paper, we use PESQ and STOI and wakeup rate as evaluation metrics. The main purpose of the development of this algorithm is to be applied to noise reduction in kitchen scenarios of a smart home. We tested the effect of the algorithm in two scenes, All test dataset and Kitchen most, respectively. When testing PESQ, we used the above test dataset with 15,000 data. When testing the wakeup rate, we recorded the real-world data of a wakeup word with Smart Home Brain Screen[2]. The wakeup word is a two-syllable Chinese pronunciation ("xiao U xiao U"), and the test dataset of wakeup words has ~56.3k positive examples (~30h) and ~60.4k negative examples (~72h).

[2] https://www.haier.com/business/smarthome/product/znckmb/20220610_182324.shtml?from=search&spm=cn.31493_pc.product_20200325.1

Table 2. Performance of PESQ on Test Dataset

Evaluation metrics	PESQ		STOI		(Acc. %)KWS Rate	
Noise type	All test dataset	Kitchen most	All test dataset	Kitchen most	All test dataset	Kitchen most
Origin	1.55	1.69	0.60	0.64	84.5	87.1
IMCRA&OMLSA[4]	1.668	1.94	0.63	0.69	89.1	89.9
LSTM out	2.03	2.11	0.71	0.73	92.3	93.7
TDCRN out	2.18	2.37	0.75	0.76	94.5	96.4
TDCRN & OMLSA	**2.32**	**2.64**	**0.82**	**0.85**	**95.3**	**97.6**

PESQ on the test dataset was calculated with different algorithms respectively, as shown in Table 2. Due to the concentration of the test, the short-time environmental noise which is smooth accounts for a relatively small percentage, mostly the type of noise with large power spectrum variation and short duration, which IMCRA cannot accurately estimate, and the enhancement of PESQ and STOI is minimized. The direct output of TDCRN has better performance than that of LSTM, indicating that the inclusion of CED results is favorable to complex target training. Although the direct output of the model improves PESQ and STOI to some extent, the speech quality can be further enhanced after being processed with OM-LSA.

Using the same wakeup engine, we set the false alarm to 1 time/24 hours, and distribute the wakeup rate of sound signals processed with different algorithms. Due to the poor performance in noise reduction of IMCRA, the result of wakeup rate is also unqualified. Although the direct output of the model can improve wakeup rate to some extent, it will lead to the loss, distortion, and aberration of noise speech. After being processed by OM-LSA, the wakeup rate can be continuously improved.

Table 3. PESQ on Test Dataset with different SNR

Evaluation metrics	-5dB	0dB	5dB	10dB	Avg.
Origin	2.20	2.31	2.36	2.52	2.35
IMCRA&OMLSA[4]	2.25	2.39	2.56	2.74	2.49
LSTM out	2.32	2.57	2.72	2.96	2.64
TDCRN out	2.38	2.62	2.77	3.03	2.70
TDCRN & OMLSA	**2.47**	**2.70**	**2.85**	**3.10**	**2.78**

PESQ on the test dataset with different SNR was calculated with different algorithms respectively, as shown in Table 3. It can be found that the performance of TDCRN&OMLSA is better than that of baseline IMCRA&OMLSA and LSTM, which proves the effectiveness of signal processing method combined with deep learning method.

Table 4. Calculation Cost of different algorithms

Evaluation metrics	Para.(M)	Time(ms)
IMCRA&OMLSA[4]	-	0.09
LSTM out	0.27	0.34
TDCRN out	0.29	0.31
TDCRN & OMLSA	-	0.36

With 2.90GHz i7-10700 CPU as the test machine, we calculate the average computation time of 200 audio files processed by different algorithms. Each file is 4 seconds long with 250 frames. As shown in Table 4, TDCRN runs 0.03ms faster than LSTM, indicating that although TDCRN parameters are slightly more than LSTM, its computation is slightly less than LSTM. The average time of TDCRN & OMLSA is 0.36ms per frame, which can be processed in real-time.

It is worth noting that the model proposed in this paper only supports 16000Hz data, and its effective frequency range is 0-8000Hz, which meets most use scenarios. For audios with other sample rates, we need to modify the model input dimension or resample the original data, which will increase the workload of model adaptation. Later, we consider that the model can be applied to data with different sample rates.

6. Conclusions

In this paper, we introduce deep learning into the estimation of noise spectrum, and use the conventional method of signal processing to estimate the spectrogram gain value of noise reduction, which can be used to suppress noises with greater fluctuations of the power spectrum and shorter durations. The TDCRN model implements one-dimensional convolution in the frame direction and achieves information exchange among frequencies by setting the number of output channels, which has better performance with fewer parameters. OMLSA&TDCRN that we proposed can achieve better performance than other algorithms in terms of PESQ and wakeup rate while satisfying the condition of real-time processing. In the future, we will deploy the proposed algorithm on smart devices, and also consider using the TDCRN model to improve the ability of noise rejection under reverberation conditions.

References

[1] P. C. Loizou, *Speech Enhancement: Theory and Practice*, 2nd ed. ed. Boca Raton, FL, USA: CRC, 2013.

[2] D.Wang and J. Lim, "The unimportance of phase in speech enhancement," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 30, no. 4, pp. 679-681, 1982.

[3] I. Cohen and B. Berdugo, "Speech enhancement for non-stationary noise environments," *Signal Processing.*, vol. 81, no. 11, pp. 2403-2418, 2001.

[4] I. 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.

[5] Y. Xu, J. Du, L. R. Dai, C. H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Processing Letters.*, vol. 21, no. 1, pp. 65-68, 2001.

[6] J. Heymann, L. Drude and R. Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 196-200.

[7] Y. Wang and D. Wang, "Towards Scaling Up Classification-Based Speech Separation," *IEEE Transactions on Audio Speech & Language Processing*, vol. 21, no. 7, pp. 1381-1390, 1013.

[8] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 7092-7096.

[9] Y. Hu, Y. Liu, S. Lv, M. Xing, S. Zhang, Y. Fu, J. Wu, B. Zhang and L. Xie, "Dccrn: Deep complex convolution recurrent network for phase-aware speech enhancement," *arXiv preprint arXiv*:*2008.00264*, 2020.

[10] 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," *arXiv preprint arXiv*:*1903.03107*, 2019.

[11] H.-S. Choi, S. Park, J.-H. Lee, H. Heo, D. Jeon and K. Lee, "Real-time Denoising and Dereverberation with Tiny Recurrent U-Net," *arXiv preprint arXiv*:*2102.03207*, 2021.

[12] K. Tan and D. Wang, "A convolutional recurrent neural network for real-time speech enhancement." in *Interspeech*, vol. 2018, 2018, pp. 3229–3233.

[13] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," in *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 33, no. 2, pp. 443-445, April 1985.

[14] J.-M. Valin, "A Hybrid DSP/Deep Learning Approach to Real-Time Full-Band Speech Enhancement," *arXiv preprint arXiv*:*1709.08243*, 2018.

[15] A. Li, W. Liu, X. Luo, C. Zheng and X. Li, "ICASSP 2021 Deep Noise Suppression Challenge: Decoupling Magnitude and Phase Optimization with a Two-Stage Deep Network," *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6628-6632.

[16] D. Snyder, G. Chen and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," *arXiv preprint arXiv*:*1510.08484v1*, 2015.

[17] H. Bu, J. Du, X. Na, B. Wu and H. Zhang, "AISHELL-1: An Open-Source Mandarin Speech Corpus and A Speech Recognition Baseline," *arXiv preprint arXiv*:*1709.05522v1*, 2017.