Fusing Multiple Bandwidth Spectrograms for Improving Speech Enhancement

Hao Shi*, Yuchun Shu†, Longbiao Wang†, Jianwu Dang†, Tatsuya Kawahara*
* Graduate School of Informatics, Kyoto University, Sakyo-ku, Kyoto, Japan
† Tianjin Key Laboratory of Cognitive Computing and Application,
College of Intelligence and Computing, Tianjin University, Tianjin, China
‡ Japan Advanced Institute of Science and Technology, Ishikawa, Japan
E-mail: shi@sap.ist.i.kyoto-u.ac.jp

***Abstract*—The spectrogram is a common feature of frequency domain speech enhancement (SE). It can be divided into wideband and narrowband according to the resolution of the spectrogram, which is controlled by the length of framing time. Although narrowband and wideband spectrograms have their own spectral characteristics, SE systems conventionally utilize single narrow bandwidth spectrograms. In this paper, we propose an SE system that simultaneously utilizes multiple bandwidth spectral information, more specifically, augments the wider bandwidth (16ms and 8ms) spectrograms as auxiliary information. Multiple bandwidth information fusion is implemented in the encoder in two ways: fusion only in the last layer (MI-F) and fusion layer by layer (MI-L). Experiments using the VB dataset show that different bandwidth spectrograms can provide supplementary information, which provides more than 0.1 PESQ improvement. The embedding dimension affects the position of the fusion position: MI-F requires less embedding dimension, while MI-L requires a larger dimension and more varied bandwidth. Moreover, the spectrogram which differs more from the main enhancement spectrogram provides better auxiliary information.**

Index Terms: speech enhancement, narrowband spectrogram, wideband spectrogram

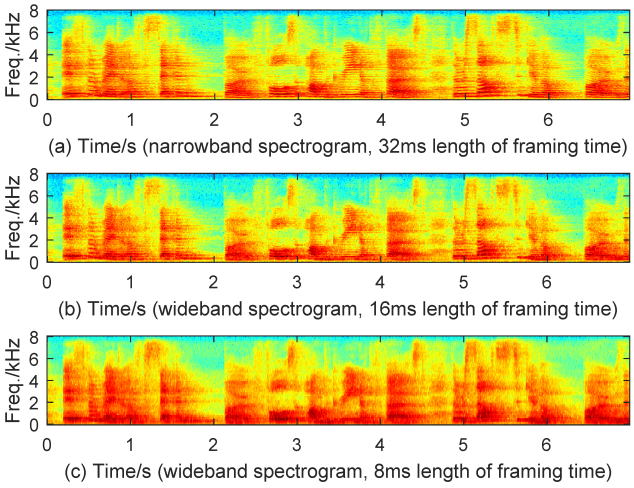

Fig. 1. Spectrogram examples extracted with different window lengths: (a) 32ms narrowband spectrogram; (b) 16ms wideband spectrogram; (c) 8ms wideband spectrogram.

I. Introduction

Noise has a great negative effect on speech signal processing [1]. As speech applications become popular, it is necessary to improve their performance in noisy scenarios [2]. Speech enhancement (SE) [2] is dedicated to recovering clean speech from noisy speech signals. Traditional SE methods [3], [4], [5], [6] are based on some established prior assumptions. In addition, these methods rely on the parameter setting and manual tuning. With the development of deep learning [7], many studies show that deep learning-based SE [8], [9], [10], [11] performs better than the traditional methods. Among these deep learning-based SE methods [12], [13], [14], [15], the frequency-domain enhancement methods are still widely used.

Spectrogram is a common feature for frequency-domain SE [16], [17], [18]. According to the resolution of the spectrogram, which is controlled by the length of framing time, it can be divided into wideband and narrowband [19]. The two kinds of spectrograms are much different and have their own characteristics [19]. Fig. 1 shows the spectrograms extracted by 8ms, 16ms, and 32ms length of framing time. Because of the short time period of each frame, wideband spectrograms have better time resolutions and can capture the rapid amplitude changes [20]. In the wideband spectrograms, the formant information of speech can be clearly seen, but the harmonic frequencies cannot be seen [20]. On the other hand, the narrowband spectrograms have longer frame lengths. It is too long to capture the rapid changes in amplitude [20], but have better spectral resolutions. It is easy to see the position of the harmonics in the narrowband spectrograms, but difficult to spot the position of the formant [20].

Although there is information complementarity between spectrograms with different bandwidths, the current SE system conventionally uses spectrograms extracted by a single window length as input and output. Some related works use convolutional neural network to extract multi-scale features [21], [22], [23] instead of multiple bandwidth spectrogram inputs.

In this paper, we design a multiple input SE system by incorporating 8ms and 16ms bandwidth spectrogram to the 32ms bandwidth spectrogram enhancement system. Spectro-

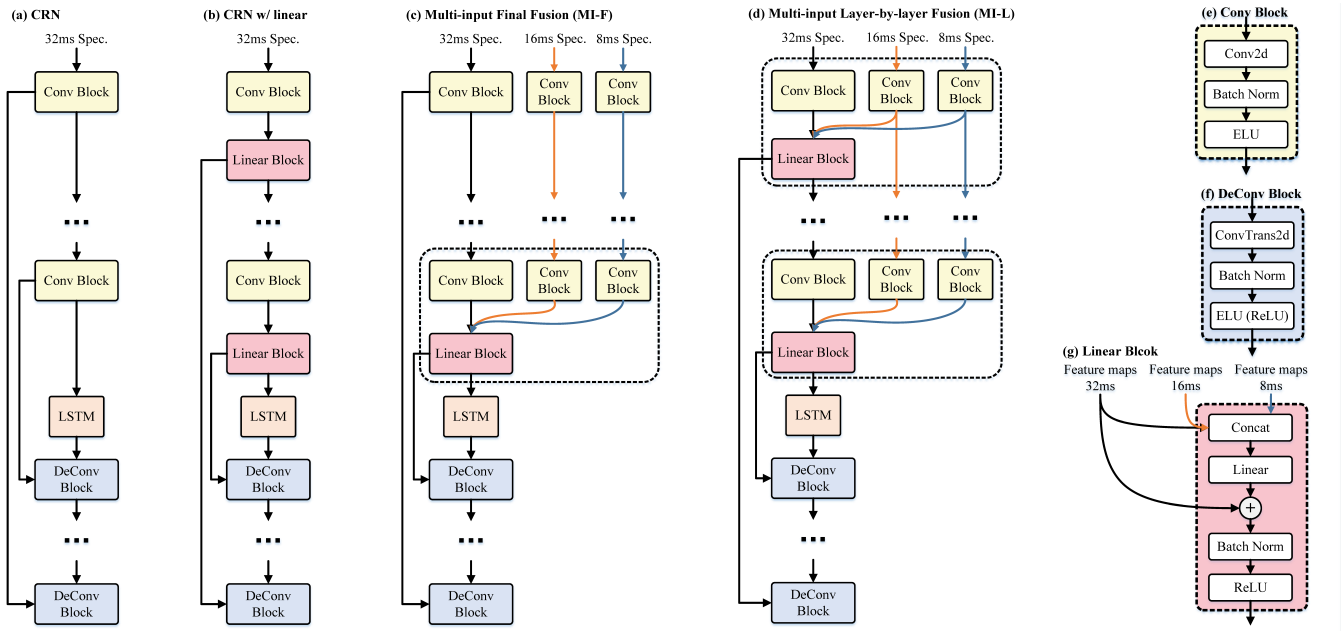

Fig. 2. Flowchart of (a) CRN; (b) CRN with Linear Blocks; (c) Multi-input Final Fusion (MI-F); (d) Multi-input Layer-by-layer Fusion (MI-L); (e) Structure of Conv Block; (f) Structure of DeConv Block; (g) Structure of Linear Block.

grams of different bandwidths are processed by multiple convolution blocks separately, and they are fused in the encoder. The difference between the two proposed methods is in the fusion position. More specifically, different bandwidth spectrograms are fused only in the last encoder layer (MI-F) or layer by layer (MI-L). We propose to use Linear Blocks to fuse different information. For MI-F, one Linear Block is only added to the last encoder layer; for MI-L, Linear Blocks are added after each encoder layer.

The rest of this paper is organized as follows. Section 2 describes the baseline model. Section 3 introduces our proposed methods. Section 4 presents the dataset, experimental settings, and experimental results. Section 5 gives the conclusion of this paper and future work.

II. Baseline Model

We choose Convolutional Recurrent Neural Network [24] (CRN, shown in Fig. 2–(a)), which performs well in frequency-domain SE as a baseline system. It contains an encoder:

$$e = \boldsymbol{E}(x) \tag{1}$$

where x and e are the noisy input spectrogram and the output of the encoder, respectively. $\boldsymbol{E}$ is the encoder of CRN, which contains several Conv Blocks (shown in Fig. 2–(e)). The output of the encoder is fed into the LSTM layers:

$$l = \boldsymbol{L}(e) \tag{2}$$

where l is the output of the LSTM layers. Then, l is input to the decoder:

$$m = \boldsymbol{D}(l) \tag{3}$$

where m is the output of decoder. $\boldsymbol{D}$ is the decoder of CRN, which contains several DeConv Blocks (shown in Fig. 2–(f)).

In this paper, we adopt a masking-based SE system:

$$\hat{o} = m * x \tag{4}$$

where $\hat{o}$ is the final enhanced spectrogram. When training the network, we use the signal approximation (SA) [25], [26]. The loss function of training is as follow:

$$\mathcal{L}_{SA} = \frac{1}{tf}\sum\nolimits_{t,f} ||\hat{o} - c||_F^2, \tag{5}$$

where t, f represent time and frequency respectively, and c is the clean spectrogram.

III. Proposed Method

In this paper, we utilize supplementary information of different bandwidth spectrograms. The proposed method inputs multi-bandwidth spectrograms simultaneously.

A. Structure of Neural Network

The flowcharts of the proposed methods are shown in Fig. 2–(c) and Fig. 2–(d). Both of Multi-input Final Fusion (**MI-F**) and Multi-input Layer-by-layer Fusion (**MI-L**) have an encoder, LSTM layers and a decoder. The network structure in front of the LSTM layers comprises the encoder. We use a Linear Block (shown in Fig. 2–(g)) to fuse the information of multiple bandwidth spectrograms:

$$h = \boldsymbol{LB}(fm_{32}, fm_{16}, fm_8) \tag{6}$$

where the fm_{32}, fm_{16}, fm_8 are feature maps of 32ms, 16ms, and 8ms bandwidth spectrograms respectively. $\boldsymbol{LB}$ represents the Linear Block, and h is the output of Linear Block. h and fm_{32} have the same feature dimension, which is realized by the linear layer of the Linear Block. For **MI-F**, Linear Block is

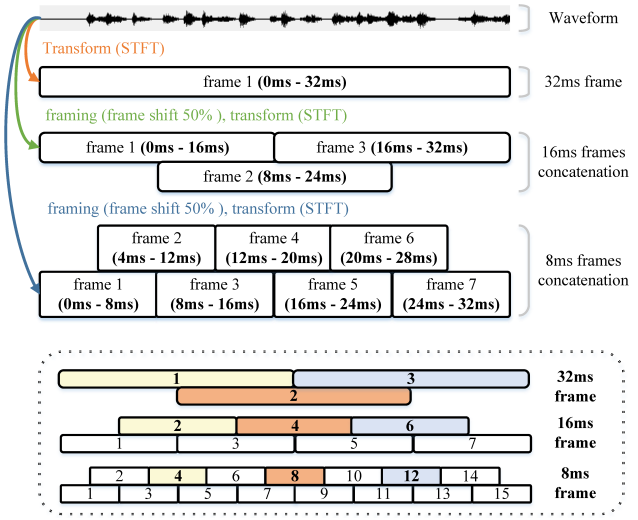

Fig. 3. 16ms and 8ms features aligned with 32ms features for framing

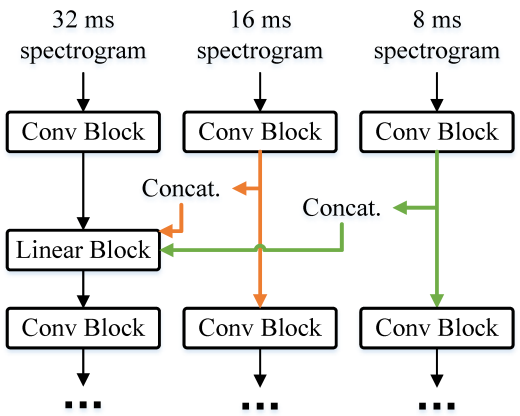

Fig. 4. Diagram of the frame concatenation.

only added to the last layer of the encoder. For **MI-L**, Linear Blocks are used to fuse the multiple bandwidth information after each Conv Block in the encoder. The residual connection is used between the corresponding encoder layer and the decoder layer. For layers without a Linear Block, we directly input the output of the Conv Block into the corresponding layer of the decoder. When there is a Linear Block, we input the output of the Linear Block to the corresponding layer of the decoder. The proposed network can be expressed as follows:

$$m = \boldsymbol{N_{MI-F}}(fm_{32}, fm_{16}, fm_8), \tag{7}$$

or

$$m = \boldsymbol{N_{MI-L}}(fm_{32}, fm_{16}, fm_8), \tag{8}$$

where $\boldsymbol{N_{MI-F}}$ and $\boldsymbol{N_{MI-L}}$ are networks of proposed MI-F and MI-L methods. The final enhanced spectrogram can be obtained by Eq. (4).

B. Processing of Input Features

Spectrograms extracted with different time periods have different information in the same time frame. With different lengths of framing time and frame shift, the frame number and information of each frame are also different. In order to ensure that the corresponding frames of different bandwidth spectrograms are aligned when input to the network, we concatenate adjacent frames of 16ms and 8ms spectrograms. This process is applied after the Conv Block and before the Linear layer. In this work, the frame shift was 50%. One frame of 32ms spectrogram corresponds to adjacent 3 frames of 16ms spectrogram; one frame of 32ms spectrogram corresponds to adjacent 7 frames of 8ms spectrogram. In addition, to align the frames, the start and end time of the 32ms frame must be the same as that of 16ms/8ms after framing. This means that the i-th 32ms frame corresponds to the framing centered on the $2i$-th 16ms frame and the corresponding framing centered on the $4i$-th 8ms frame. The corresponding relationship is shown in Fig. 3. The diagram of the frame concatenation is shown in Fig. 4.

C. Training of the Network

The network takes SA masking as a learning target which calculates the loss with Eq. (5). The output of the network is the mask m for the 32ms spectrogram, which is used for enhancement in Eq. (4)

IV. Experiments

We used a public VB dataset[1], which is synthesized from the Voice Bank dataset and the Demand dataset. It contains training and test sets. We selected all data of two speakers (one male and one female) as the validation set. This will ensure that the test speakers were unseen. Finally, the training set contained 10,705 utterances, and the validation set contained 867 utterances. We used the best-performing model under the validation set for evaluation. The test set contained 824 utterances in total. The sampling rate of the original dataset is 48k Hz. We downsampled the audio to 16k Hz in our experiments. For feature extraction, we used the following parameters to extract 32ms spectrogram: window length was 512; hop length was 256; short-time Fourier transform points was 512. For 16ms/8ms spectrograms, these hyperparameters were set to 256/128, 128/64, 256/128. We used the magnitude of the spectrogram as both input and output of the experiments.

All models contain a 5-layer Conv Block encoder and a 5-layer DeConv Block decoder. The parameters of the convolutional layer in the Conv Block are as follows: kernel size of (3,2), stride of (2, 1) and padding of (0, 1). The parameters of the deconvolutional layer in the DeConv Block are as follows: kernel size of (3,2), stride of (2, 1) and padding of (0, 0) except that (1, 0) was used for the 4th layer; the activation function of the last layer is ReLU, and the other layers are ELU. The numbers of feature maps in the encoder

[1]https://datashare.ed.ac.uk/handle/10283/2791

TABLE I

RESULTS OF DIFFERENT ENHANCEMENT SYSTEMS: 8MS (16MS, 32MS) FEAT. REPRESENTS THAT 8MS (16MS, 32MS) FEATURE AS INPUT AND OUTPUT FEATURE; 8MS (16MS) AUX. REPRESENTS THAT THE AUXILIARY FEATURE IS 8MS (16MS); 8, 16MS AUX. REPRESENTS THAT THE AUXILIARY FEATURES ARE BOTH 8MS AND 16MS SPECTROGRAMS.

SYSTEMS		SIG	BAK	OVRL	PESQ
noisy (original)		3.35	2.44	2.63	1.970
CRN	8ms feat.	3.61	2.92	2.92	2.264
	16ms feat.	3.62	3.07	3.02	2.481
	32ms feat.	3.51	2.98	3.02	2.563
	+ linear	3.56	3.14	3.01	2.502
MI-F	8ms aux.	3.69	**3.25**	3.16	2.657
	16ms aux.	3.61	3.14	3.07	2.568
	8, 16ms aux.	3.54	3.20	3.03	2.563
MI-L	8ms aux.	3.51	3.19	3.03	2.607
	16ms aux.	3.71	3.18	3.13	2.593
	8, 16ms aux.	**3.81**	3.22	**3.22**	**2.662**

was $1 \rightarrow 16 \rightarrow 32 \rightarrow 64 \rightarrow 128 \rightarrow 256$, and the numbers of feature map in the decoder were $512 \rightarrow 256 \rightarrow 128 \rightarrow 64 \rightarrow 32 \rightarrow 1$. A Linear Block contained one linear layer.

For baseline methods, we tried **32ms**, **16ms**, and **8ms** spectrogram as input features for CRN. With different input feature dimensions, the dimensions of multiple bandwidth spectrograms will also have different dimensions after the convolutional processing, which will affect the number of nodes in the LSTM layers. For the input of 32ms spectrogram 1,792 LSTM layer nodes were used; 768 nodes for 16ms spectrogram and 256 nodes for 8ms spectrogram. All models had two LSTM layers. In order to make a fair comparison by considering the effect of the Linear Block, we add Linear Blocks after each Conv Block for the 32ms spectrogram baseline (**+ linear**), which is shown in Fig. 2–(b).

To evaluate the performance of each method, we used SIG (values range from 1 to 5, higher value indicates clearer and more natural with less degradation)[27], BAK (values range from 1 to 5, higher value indicates less intrusive of background noise)[27], OVRL ([1=bad, 2=poor, 3=fair, 4=good, 5=excellent])[27] and the perceptual evaluation of speech quality (PESQ) [27], [28].

A. Effect of Different Bandwidth

Table I shows the results of different SE systems. SE systems were greatly affected by the bandwidth of input and output features. Compared with the "16ms" and "8ms" systems, the "32ms" system obtains the best PESQ. With the increase of the bandwidth, the PESQ score tends to decrease. However, the wideband systems had the better speech signal recovery according to SIG, but the "8ms" system had the worst performance in suppressing intrusion noise (BAK) and overall signal recovery (OVRL). Due to the transient nature, the speech signal is periodic in the range of vowels. The "8ms" spectrogram is too short to cover transient stability, thus the "8ms" system had the worst performance.

TABLE II

THE INPUT DIMENSION (32MS, 16MS, 8MS) OF LINEAR BLOCK IN DIFFERENT ENCODER LAYERS: WE USE THE OUTPUT DIMENSION OF CONV BLOCK (n) $\times$ THE NUMBER OF FRAMING m.

Encoder Layers	32ms	16ms	8ms
1	128×1	64×3	32×7
2	63×1	31×3	15×7
3	31×1	15×3	7×7
4	15×1	7×3	3×7
5	7×1	3×3	1×7

B. Effect of Linear Block

We directly added a Linear Block to the 32ms-based system for fair comparisons. A Linear Block was added after each Conv Block in the encoder without introducing auxiliary information of other bandwidths. The results in Table I show that adding Linear Blocks slightly improved SIG and BAK scores. However, OVRL and PESQ of the enhanced speech signal are degraded.

C. Effect of MI-F

In the MI-F method, a Linear Block is added to the last layer of the encoder. The experimental results in Table I show that the best performance was obtained when using the "8ms aux.". With "16ms aux." and "8, 16ms aux.", SIG, BAK, and OVRL were improved but the improvement of PESQ was limited. The results show a trend that "8ms aux." was better than "16ms aux.", and "16ms aux." was better than "8, 16ms aux.". We reason that it is difficult for a single linear layer to incorporate a lot of information. Table II shows the input dimension of the Linear Block in different encoder layers. The 16ms spectrogram contains 9 dimensions (3×3) in the fifth encoder layer, while there are only 7 dimensions (1×7) for the 8ms spectrogram. High-dimensional (9-dimensional embedding for 16ms; 16-dimensional embedding for 8, 16ms aux.) features are not well fused by the single linear layer, resulting in a limited performance improvement.

D. Effect of MI-L

In the MI-L method, a Linear Block is added after each Conv Block in the encoder for information fusion. The experimental results in Table I show that the best performance was obtained when using the "8, 16ms aux.", while "8ms aux." and "16ms aux." had limited improvement for PESQ. When 8ms and 16ms spectrograms were used into the network as auxiliary information simultaneously, all evaluation measures were greatly improved. This shows that with layer-by-layer fusion the different spectral information was fused well.

E. Difference Between MI-F and MI-L

In both MI-F and MI-L, "8ms aux." achieved better performance than "16ms aux.". Compared with the 16ms spectrogram, the 8ms spectrogram has a larger difference from the 32ms spectrogram. Therefore, spectral information with larger differences is more effective. In addition, with sufficient fusion capability, more information can lead to

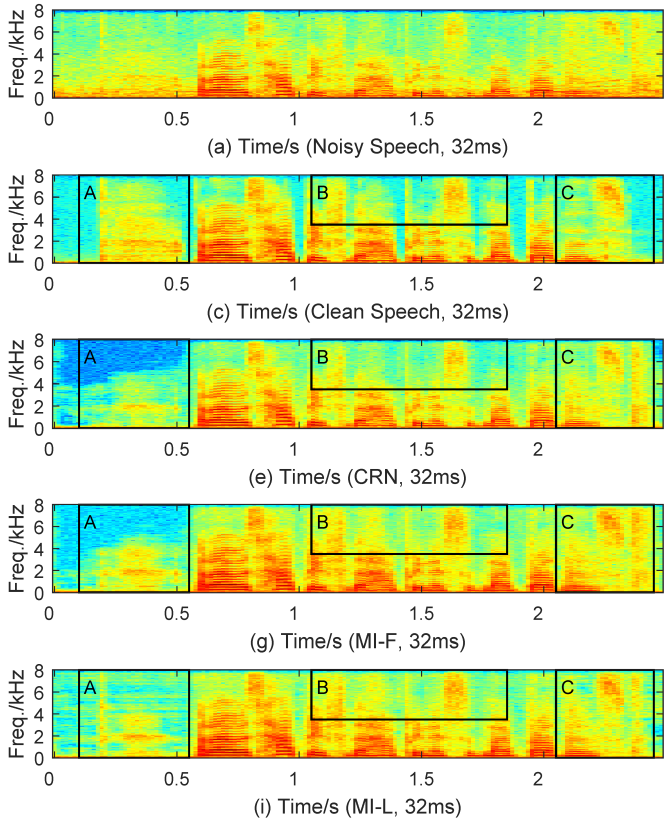

Fig. 5. Spectrograms of different SE systems.

better performance. MI-L outperforms MI-F on all evaluation measures. Besides, MI-F needs the fusion layer to have a smaller dimension, while MI-L needs the fusion layer to have a larger dimension. Furthermore, the auxiliary spectrogram of MI-F needs to be much different from the main enhanced spectrogram, while auxiliary spectrograms of MI-L are required to have more complete information.

F. Effect of Proposed Methods on Spectrogram

Fig. 5 shows the spectrograms of different SE systems. The main difference between these SE methods is the restoration of high frequencies and the processing of silent segments. Part A is a silent segment, "CRN" lost a lot of energy, while "MI-L" has better signal recovery. Furthermore, both "MI-F" and "MI-L" achieved recovery of sharper high-frequency detail. For Part B, "MI-F" and "MI-L" had better high-frequency recoveries than "CRN". For Part C, some noise was not removed in all enhanced spectrograms, but "MI-L" contains less noise. We reason that the time-varying information provided by the wideband spectrogram helps narrowband spectrogram restoration. Furthermore, although the PESQ of "MI-F" was the same as that of "MI-L", there is still some information loss in "MI-F". Spectrograms with more bandwidth as input features help preserve spectral information.

V. Conclusions

In this paper, we aim to improve a narrowband-based SE system with the wider bandwidth spectrograms as auxiliary information. We propose multi-input final fusion (MI-F) and multi-input layer-by-layer fusion (MI-L) to incorporate information from different bandwidth spectrograms. MI-F adds a Linear Block only to the last layer of the encoder, while MI-L adds Linear Block after each Conv Block in the encoder for information fusion. With better fusion ability, MI-L achieves a better performance. Moreover, systems with larger differences in bandwidth achieve better performance. The proposed methods achieved better spectral recovery on silent segments and high-frequency spectrograms.

References

[1] M. K. Pichora-Fuller, B. A. Schneider, and M. Daneman, "How young and old adults listen to and remember speech in noise," *The Journal of the Acoustical Society of America*, vol. 97, no. 1, pp. 593–608, 1995.

[2] P. C. Loizou, *Speech enhancement: theory and practice*. CRC press, 2007.

[3] J. Meyer and K. Simmer, "Multi-channel speech enhancement in a car environment using wiener filtering and spectral subtraction," in *Proc. ICASSP*, vol. 2, 1997, pp. 1167–1170 vol.2.

[4] M. Hasan, S. Salahuddin, and M. Khan, "A modified a priori snr for speech enhancement using spectral subtraction rules," *IEEE Signal Processing Letters*, vol. 11, no. 4, pp. 450–453, 2004.

[5] R. Martin, "Speech enhancement using mmse short time spectral estimation with gamma distributed speech priors," in *Proc. ICASSP*, vol. 1, 2002, pp. I–253–I–256.

[6] Y. Ephraim and H. Van Trees, "A signal subspace approach for speech enhancement," in *Proc. ICASSP*, vol. 2, 1993, pp. 355–358 vol.2.

[7] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[8] Y. Xu, J. Du, L.-R. Dai, and 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, 2014.

[9] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder." in *Proc. Interspeech*, vol. 2013, 2013, pp. 436–440.

[10] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM TASLP*, vol. 23, no. 1, pp. 7–19, 2015.

[11] A. D¨ifossez, G. Synnaeve, and Y. Adi, "Real Time Speech Enhancement in the Waveform Domain," in *Proc. Interspeech*, 2020, pp. 3291–3295.

[12] H. Shi, L. Wang, M. Ge, S. Li, and J. Dang, "Spectrograms fusion with minimum difference masks estimation for monaural speech dereverberation," in *Proc. ICASSP*, 2020, pp. 7544–7548.

[13] M. H. Soni, N. Shah, and H. A. Patil, "Time-frequency masking-based speech enhancement using generative adversarial network," in *Proc. ICASSP*, 2018, pp. 5039–5043.

[14] A. Pandey and D. Wang, "A new framework for CNN-based speech enhancement in the time domain," *IEEE/ACM TASLP*, vol. 27, no. 7, pp. 1179–1188, 2019.

[15] 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," *Proc. Interspeech*, pp. 2472–2476, 2020.

[16] N. Zheng and X.-L. Zhang, "Phase-aware speech enhancement based on deep neural networks," *IEEE/ACM TASLP*, vol. 27, no. 1, pp. 63–76, 2019.

[17] D. Takeuchi, K. Yatabe, Y. Koizumi, Y. Oikawa, and N. Harada, "Effect of spectrogram resolution on deep-neural-network-based speech enhancement," *Acoustical Science and Technology*, vol. 41, no. 5, pp. 769–775, 2020.

[18] L. Sun, J. Du, L.-R. Dai, and C.-H. Lee, "Multiple-target deep learning for LSTM-RNN based speech enhancement," in *Proc. HSCMA*, 2017, pp. 136–140.

[19] S. Cheung and J. Lim, "Combined multi-resolution (wide-band/narrowband) spectrogram," in *Proc. ICASSP*, 1991, pp. 457–460 vol.1.
[20] A. V. Oppenheim, "Speech spectrograms using the fast fourier transform," *IEEE Spectrum*, vol. 7, no. 8, pp. 57–62, 1970.
[21] Y. Koizumi, N. Harada, and Y. Haneda, "Trainable adaptive window switching for speech enhancement," in *Proc. ICASSP*, 2019, pp. 616–620.
[22] C. Xu, W. Rao, E. S. Chng, and H. Li, "Spex: Multi-scale time domain speaker extraction network," *IEEE/ACM TASLP*, vol. 28, pp. 1370–1384, 2020.
[23] X. Xiang, X. Zhang, and H. Chen, "A convolutional network with multi-scale and attention mechanisms for end-to-end single-channel speech enhancement," *IEEE Signal Processing Letters*, vol. 28, pp. 1455–1459, 2021.
[24] K. Tan and D. Wang, "A convolutional recurrent neural network for real-time speech enhancement," *Proc. Interspeech*, pp. 3229–3233, 2018.
[25] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM TASLP*, vol. 26, no. 10, pp. 1702–1726, 2018.
[26] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. L. Roux, J. R. Hershey, and B. Schuller, "Speech enhancement with lstm recurrent neural networks and its application to noise-robust ASR," in *Proc. International conference on latent variable analysis and signal separation*. Springer, 2015, pp. 91–99.
[27] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE TASLP*, vol. 16, no. 1, pp. 229–238, 2008.
[28] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. ICASSP*, vol. 2, 2001, pp. 749–752 vol.2.