Monaural Speech Enhancement Based on Spectrogram Decomposition for Convolutional Neural Network-sensitive Feature Extraction

Hao Shi[1], *Longbiao Wang*[2,*], *Sheng Li*[3], *Jianwu Dang*[2,4], *Tatsuya Kawahara*[1,*]

[1]Graduate School of Informatics, Kyoto University, Kyoto, Japan
[2]Tianjin Key Laboratory of Cognitive Computing and Application,
College of Intelligence and Computing, Tianjin University, Tianjin, China
[3]National Institute of Information and Communications Technology, Kyoto, Japan
[4]Japan Advanced Institute of Science and Technology, Ishikawa, Japan

shi@sap.ist.i.kyoto-u.ac.jp

Abstract

Many state-of-the-art speech enhancement (SE) systems have recently used convolutional neural networks (CNNs) to extract multi-scale feature maps. However, CNN relies more on local texture than global shape, which is more susceptible to degraded spectrogram and may fail to capture the detailed structure of speech. Although some two-stage systems feed the first-stage enhanced and original noisy spectrograms to the second stage simultaneously, this does not guarantee sufficient guidance for the second stage since the first-stage spectrogram can not provide precise spectral details. In order to allow CNNs to perceive clear speech component boundary information, we compose feature maps with spectrograms containing evident speech components according to the mask value from the first stage. The positions corresponding to the mask greater than certain thresholds are extracted as feature maps. These feature maps make the boundary information of speech components obvious by ignoring others, thus making CNNs sensitive to input features. Experiments on the VB dataset show that with a proper decomposition numbers, the proposed method can enhance SE performance, which can provide 0.15 PESQ improvement. Besides, the proposed method is more effective for spectral detail recovery.

Index Terms: Spectrogram decomposition, speech component awareness, speech enhancement, deep learning

1. Introduction

In recent years, speech applications have become increasingly popular with their convenience. Application scenarios have subsequently become more complex, which further requires improved performance of speech front-end processing [1, 2]. The noise in the real scenarios will have a great negative impact on speech signal processing [3, 4], which makes speech noise reduction receive more and more attention. Therefore, it is important to develop a front-end processing to recover clean speech components from noisy speech signals.

Recently, deep learning-based speech enhancement (SE) systems [5, 6, 7] show better performance than the traditional signal processing methods [5, 8, 9]. Common networks are fully connected neural networks, recurrent neural networks (RNNs) [10, 11, 12] and convolutional neural networks (CNNs) [13, 14]. Different network structures have different characteristics: RNNs can capture the long-term contextual information to consider long-term acoustic information [10]; CNNs introduce convolution kernels to obtain multi-scale feature maps of input features [15, 16]. Local connections and weight sharing greatly reduce model parameters.

Many recent works achieve state-of-the-art performance with CNNs [17, 18, 19, 20]. It is generally believed that humans identify objects primarily by their shape. But CNNs tend to use color and texture to make predictions rather than shape [21]. For noisy speech, the noise will destroy the speech spectrogram structure [22], especially the texture information. In addition, due to the influence of noise, it is difficult to see the shape of many important structures such as formants in noisy spectrograms. This brings difficulties to SE with CNNs. Some two-stage systems utilize the first stage to obtain enhanced spectrogram, which are then fed into the second stage as input features. Although this is helpful for network learning, the enhanced spectrogram obtained in the first stage often has great defects in details and information retention, which makes it difficult to obtain a greater improvement through the second stage.

In this paper, we address the above problem by highlighting speech components. We extract the strong speech part and ignore others, so as to make the boundary of the speech component obvious. Strong speech part is determined based on the output mask by a trained masking-based SE system. The mask value shows the proportion of speech components present. We extract the spectral information corresponding to the position where the mask is larger than a certain threshold to form a feature map. The stacking of feature maps of strong speech components enables the input features to provide sufficient speech boundary information, making the CNNs more sensitive to the input features.

The rest of this paper is organized as follows. Section 2 describes the masking-based SE. Section 3 describes our proposed method. Section 4 gives the experimental settings and results. Section 5 gives the conclusion and future work.

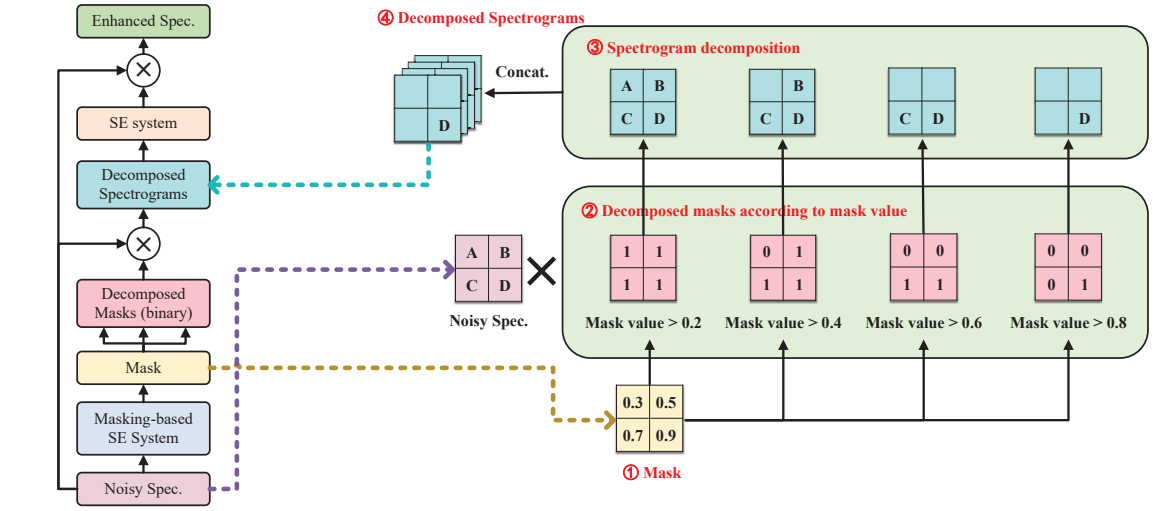
2. Related Works

Speech enhancement (SE) aims to recover clean speech from noisy speech signals. Masking-based SE systems have received more and more attention in recent years. They can be formulated as follows:

$$|\widehat{\mathbf{M}}| = \mathcal{N}(|\mathbf{Y}|), \tag{1}$$

where $\mathcal{N}$, $|\mathbf{Y}|$, $|\widehat{\mathbf{M}}|$ are neural network, noisy spectrogram and estimated mask respectively. The estimated mask shows how much speech components exist in each time–frequency (T–F)

*: Corresponding author



(a): Flowchart of proposed system **(b): One example of spectrogram decomposition**

Figure 1: *(a) Flowchart of proposed spectrogram decomposition method. (b) one example of spectrogram decomposition.*

bin. The masking-based SE uses the mask to extract the speech components in T–F bins. The enhanced spectrogram is obtained by multiplying the noisy spectrogram with the estimated mask:

$$|\widehat{\mathbf{X}}| = |\widehat{\mathbf{M}}| \odot |\mathbf{Y}|, \tag{2}$$

where the $|\widehat{\mathbf{X}}|$ is the enhanced spectrogram. Signal approximation (SA) [23] is a masking method approximated with the speech signal. It does not directly calculate the loss between the estimated and ideal masks, but uses the loss between the estimated and clean spectrogram:

$$\mathcal{L}_{SA} = \frac{1}{TF} \sum\nolimits_{T,F} \left(|\widehat{\mathbf{X}}| - |\mathbf{X}| \right)^2 \tag{3}$$

where T, F denote the time and frequency, respectively. And $|\mathbf{X}|$ denotes the clean spectrogram. The phase from the noisy speech signal will be used to reconstruct the ehanced waveform.

The two stage model contains two SE modules. Enhanced spectrogram $|\widehat{\mathbf{X}}|$ can be obtained from the first stage. Then, another SE module is employed to get the final enhanced feature:

$$|\widehat{\mathbf{M}}| = \mathcal{N}(|\widehat{\mathbf{X}}|), \tag{4}$$

$$|\widehat{\mathbf{M}}| = \mathcal{N}(|\mathbf{Y}|, |\widehat{\mathbf{X}}|), \tag{5}$$

where $|\mathbf{Y}|$ is the noisy spectrogram. The enhanced spectrogram $|\widehat{\mathbf{X}}|$ can be directly input to the second stage with Eq.(4). Another way is to input the enhanced and noisy spectrograms to the neural network simultaneously with Eq. (5), which will ensure that the noise spectrogram compensates for the information lost in the enhanced spectrogram. Both of them get the final enhanced spectrogram by Eq. (2).

3. Proposed Method

Spectrogram is a widely used feature to SE. However, the noise greatly deteriorates the structure of speech components in the spectrogram, especially when the signal-noise ratio is small. This will greatly affect the CNNs extraction of multi-scale features with noisy spectrograms. We design spectrogram decomposition to extract input features that are beneficial to CNNs, which solve the problem that CNNs is insensitive to shape.

3.1. System Description

Fig. 1–(a) shows the flowchart of our proposed method. The proposed method has two stages. Masking-based SE is chosen to estimate a mask for the first stage. Then the mask is used to decompose noisy spectrogram. The decomposed feature $|\mathbf{D}|$ is as input feature to the second stage:

$$|\widehat{\mathbf{M}}| = \mathcal{N}(|\mathbf{D}|), \tag{6}$$

It should be emphasized that the enhanced spectrogram is not included as input feature in the second stage. Both two stages adopt the structure of convolutional recurrent neural network (CRN) [18]. It is a U-Net-based network. It contains a encoder, LSTM layers and a decoder. The encoder has some convolutional block to extract multi-scale feature maps. The LSTM layers are used to get better deep embedding than the output of encoder. The decoder has some deconvolutional block to restore features.

Algorithm 1: Pseudo-code of the spectrogram decomposition

Input: number of intervals n, mask m, noisy spectrogram $spec_n$
Output: decomposed spectrogram $spec_d$

$slices = n$;
$step = 1/slices$;
$spec_d = spec_n$;
for *i in range (1, slices)* **do**
  $x\_slice = float(bool(m > step * i))$;
  $spec_d = concatenate(spec_d, x\_slice * spec_n)$;
end

3.2. Spectrogram Decomposition-based Feature Extraction for CNN

We decompose the spectrogram according to the value of $|\widehat{\mathbf{M}}|$. A mask value shows the proportion of speech components in noisy speech. Since most of the values of mask are in $(0, 1)$, we divide $(0, 1)$ into **n** equidistant intervals. The mask value greater than the lower bound of the interval is used to form a new feature map. The speech components with larger mask values have a high probability of being prominent. Our purpose is to only retain strong speech components in the decomposed spectrograms, ignoring other information. Thus, clear edge connection can be formed, so as to highlight the shape of the speech components, which can assist CNNs in extracting

multi-scale feature maps.

This decomposition process can be divided into two steps: mask estimation and spectrogram decomposition. The i–th decomposed mask is:

$$\mathbf{m}_i^{t,f} = \begin{cases} 0, |\widehat{\mathbf{M}}|_i^{t,f} < \mathbf{b}_i^{t,f}, \\ 1, |\widehat{\mathbf{M}}|_i^{t,f} > \mathbf{b}_i^{t,f}, \end{cases} \tag{7}$$

where $\mathbf{b}_i$ denotes the lower bound of the i–th interval and $\mathbf{m}_i$ denotes the decomposed mask. Thus, we can obtain $\mathbf{n}$ decomposed masks.

We use the decomposed masks to extract the information of the corresponding position in the noisy spectrogram:

$$\mathbf{d}_i = \mathbf{m}_i \odot |\mathbf{Y}|, \tag{8}$$

where $\mathbf{d}_i$ denotes the i–th decomposed spectrogram. For a feature map with a large lower bound of the mask interval, the speech information is more obvious. Finally, we concatenate the obtained feature maps to get a multiple-channel feature:

$$|\mathbf{D}| = concat(\mathbf{d}_i), i \in [1, n], \tag{9}$$

We use the decomposed spectrograms as the input of the second stage network. It should be noted that we only use mask decomposition to obtain binary matrices instead of using the mask to enhance the spectrogram. Algorithm 7 shows the pseudo-code of the spectrogram decomposition.

4. Experiments

4.1. Experimental Settings

We used a public VB dataset for experiments[1], which is synthezied from Voice Bank dataset and the Demand dataset. It contains training and testing sets. We selected all the data of two speakers (one male and one female) as the validation set. This will ensure that the speakers were unseen. Finally, our 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, and we downsampled the audio to 16k Hz in our experiments. For feature extraction, we used the following parameters: window length was 512; hop length was 256; short-time fourier transform points was 512. We used the magnitude of the spectrogram as both input and output of this experiment.

We used the convolutional recurrent neural network (CRN) [18] in these experiments. In all experiments, except for the input dimensions, the network structures were the same. They have 5 encoder layers and 5 decoder layers. The LSTM had two layers, each layer had 1,792 nodes. We used n to represent the input feature dimensions. When training the network, we used the mean squared error as the loss function; the batch size was 18; the initial learning rate was 0.0006; the optimizer was Adam; the epoch was 50. We tested three baseline methods:

* **CRN**: the network was trained with Eq. (3); the input feature is noisy spectrogram; the input size was $1 \times 257 \times F$.
* **CRN-stack**: a two-stage method; it contains two CRNs, the input of the first CRN is noisy spectrogram, the input of the second CRN is the enhanced output from the first CRN.
* **CRN-stack-w-noisy**: the input of the second CRN is the concatenation of noisy and enhanced spectrogram; the other structures are same with "CRN-stack".

We used the perceptual evaluation of speech quality (PESQ) [24], CSIG (higher value indicates clearer and more natural speech)[24], CBAK (higher value indicates the less intrusive of background noise)[24], COVL (Overall speech quality, the higher the better)[24] as evaluation metrics.

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

Table 1: *Results of different enhancement systems.*

System	CSIG	CBAK	COVL	PESQ
noisy (input)	3.35	2.44	2.63	1.97
CRN	3.505	2.978	3.020	2.563
CRN-stack	3.596	3.036	3.095	2.617
CRN-stack-w-noisy	3.829	3.065	3.233	2.635
decomposition	**4.015**	**3.099**	**3.368**	**2.722**

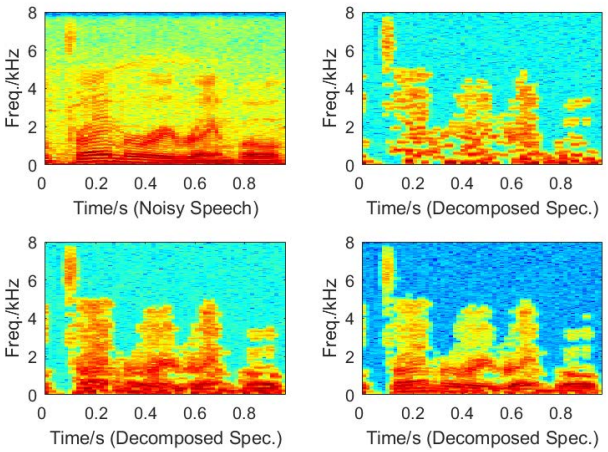


Figure 2: *Samples of decomposed spectrograms.*

4.2. Performance of Different SE Systems

Table 1 shows the results of different enhancement systems. It is difficult to get the improved results simply by stacking the network. Although the "CRN-stack" had double number of parameters, the improvement of performance was small. Even when the noisy spectrogram was added back in the second stage, the gains were still small.

In Table 1, "decomposition" denotes the proposed spectrogram decomposition-based system. The number of parameters of "decomposition" is almost the same as "CRN-stack". Compared to simple stacking, spectrogram decomposition can provide more than 0.1 PESQ improvement. Because the decomposed spectrogram is still noisy, PESQ had an 0.15 improvement from the baseline "CRN" that only takes the noisy spectrogram as input. This indicates that the speech component awareness is helpful for enhancement tasks. Moreover, "decomposition" was more effective than other methods in maintaining the speech signal, e.g., it had about 0.42 CSIG improvement from the "CRN-stack".

4.3. The Effect of Decomposed Spectrograms

Figure. 2 shows a sample of the decomposed spectrograms. We randomly selected some high value boundaries. It can be clearly seen that some speech components are highlighted. It shows that spectrogram decomposition can make the CNNs more sensitive.

4.4. Effect of Different Decomposition Numbers

Figure. 3 shows the evaluation metrics for different decomposition numbers. The decomposition number of 30 is shown in Table. 1. All decomposition methods had improved per-

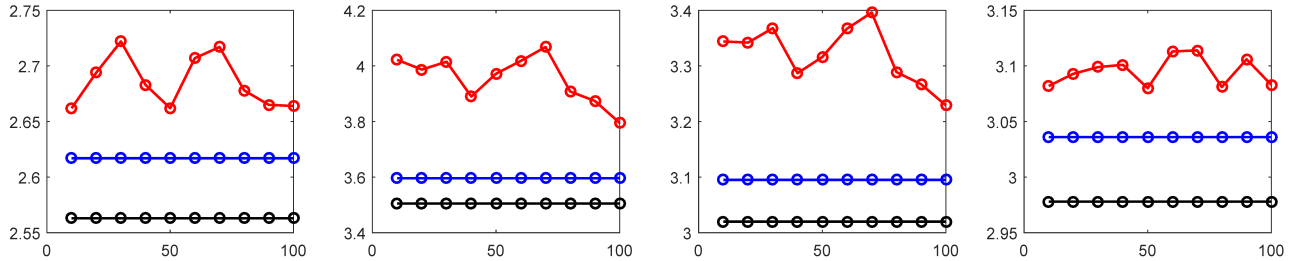


Figure 3: *Measures on different decomposition numbers: Red line represents the "decomposition"; Blue line represents the "CRN-stack"; Black line represents the "CRN".*

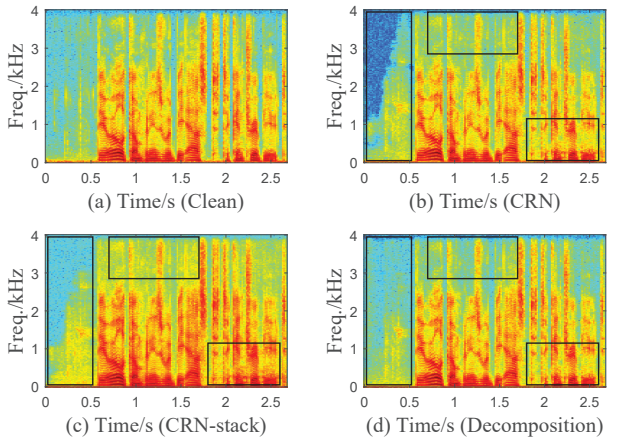


Figure 4: *Spectrograms of different enhancement systems: (a) clean speech; (b) CRN enhanced speech; (c) CRN-stack enhanced speech; (4) Decomposition enhanced speech.*

formance over baselines, especially when the decomposition numbers were 30 and 70. This trend was obvious for PESQ, CSIG, and COVL. It was not the case that finer spectrogram decomposition could lead to better enhancement performance. This implies that the appropriate decomposition number needs to be found when decomposing the spectrogram. Moreover, the CBAK was not much changed. The decompositions number was more likely to affect the recovery of speech components and the overall signal.

4.5. Effect of Spectrogram Decomposition on Spectrogram

Figure. 4 shows the spectrograms of different enhancement systems. "CRN" had serious information loss in the silent regions. Although "CRN-stack" had alleviated this problem, there was residual noise. Moreover, the energy of "CRN-stack" was greater in the speech regions, and some detailed information was lost. Compared with the other two methods, the proposed "decomposition" had better spectrogram recovery.

4.6. Effect of Spectrogram Decomposition on Feature Maps

We selected some feature maps from output of conv2d_1, which are shown in Figure. 5. The noise in the feature maps extracted by "decomposition" was greatly suppressed. In addition, the speech signal part of the feature map was also better preserved, especially the middle and high frequency parts. This shows that the proposed spectrogram decomposition can help to alleviate the robustness problem caused by texture bias in CNN.

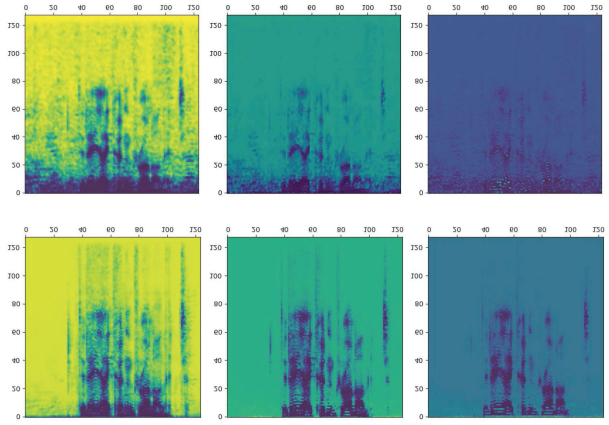


Figure 5: *Selected feature maps of baseline (CRN) and proposed decomposition method.*

For more details, please visit the URL[2].

5. Conlusion and Future Work

In this paper, we have proposed spectrogram decomposition to extract CNNs-sensetive input features for SE. We decomposed the spectrogram using the mask from a trained masking-based SE system. First, we divided the mask into equal intervals according to the value of the mask. Then the regions larger than each interval constitute a new decomposition feature map. Finally, we concatenated multiple decomposed features and input them into the network as input features. We showed that the proposed method had better speech component and overall signal recovery. Besides, a proper decomposition number can bring better enhancement performance. Moreover, the proposed spectrogram decomposition helps CNNs by extracting feature maps with less noise and prominent speech components. In the future, we will try more ways to decompose the spectrogram and combine autoML to select the decomposition number automatically.

6. Acknowledge

This work was supported by JST, the establishment of university fellowships towards the creation of science technology innovation, Grant Number JPMJFS2123. This work was supported by the National Natural Science Foundation of China under Grant 62176182.

[2]https://hshi-speech.github.io/one-example-of-feature-maps-of-spectrogram-decomposition/

7. References

[1] A. Narayanan and D. Wang, "Investigation of speech separation as a front-end for noise robust speech recognition," *IEEE/ACM TASLP*, vol. 22, no. 4, pp. 826–835, 2014.

[2] 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.

[3] Y. Gong, "Speech recognition in noisy environments: A survey," *Speech Communication*, vol. 16, no. 3, pp. 261–291, 1995.

[4] J. Benesty, S. Makino, and J. Chen, *Speech enhancement*. Springer Science & Business Media, 2006.

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

[6] R. Ellis, H. Yoo, D. Graham, P. Hasler, and D. Anderson, "A continuous-time speech enhancement front-end for microphone inputs," in *Proc. ISCAS*, vol. 2, 2002, pp. II–II.

[7] Y. Xu, J. Du, L. Dai, and C. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 65–68, 2014.

[8] 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.

[9] H. Fan, J. Hung, X. Lu, S. Wang, and Y. Tsao, "Speech enhancement using segmental nonnegative matrix factorization," in *Proc. ICASSP*, 2014, pp. 4483–4487.

[10] L. Sun, J. Du, L. Dai, and C. Lee, "Multiple-target deep learning for lstm-rnn based speech enhancement," in *Proc. HSCMA*, 2017, pp. 136–140.

[11] 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.

[12] H. Shi, L. Wang, S. Li, C. Ding, M. Ge, N. Li, J. Dang, and H. Seki, "Singing Voice Extraction with Attention-Based Spectrograms Fusion," in *Proc. Interspeech*, 2020, pp. 2412–2416.

[13] S. Fu, Y. Tsao, and X. Lu, "SNR-Aware Convolutional Neural Network Modeling for Speech Enhancement," in *Proc. Interspeech*, 2016, pp. 3768–3772.

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

[15] S. R. Park and J. Lee, "A fully convolutional neural network for speech enhancement," *arXiv preprint arXiv:1609.07132*, 2016.

[16] D. Stoller, S. Ewert, and S. Dixon, "Wave-U-Net: A Multi-Scale Neural Network for End- to-End Audio Source Separation," in *Proc. ISMIR*. Paris, France: ISMIR, Sep. 2018, pp. 334–340.

[17] A. Pandey and D. Wang, "Dense cnn with self-attention for time-domain speech enhancement," *IEEE/ACM TASLP*, vol. 29, pp. 1270–1279, 2021.

[18] K. Tan and D. Wang, "A Convolutional Recurrent Neural Network for Real-Time Speech Enhancement," in *Proc. Interspeech*, 2018, pp. 3229–3233.

[19] 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.

[20] Z. Ouyang, H. Yu, W.-P. Zhu, and B. Champagne, "A fully convolutional neural network for complex spectrogram processing in speech enhancement," in *Proc. ICASSP*, 2019, pp. 5756–5760.

[21] B. Shi, D. Zhang, Q. Dai, Z. Zhu, Y. Mu, and J. Wang, "Informative dropout for robust representation learning: A shape-bias perspective," in *Proc. ICML*, vol. 119. PMLR, 13–18 Jul 2020, pp. 8828–8839.

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

[23] Y. Liu, H. Zhang, X. Zhang, and L. Yang, "Supervised speech enhancement with real spectrum approximation," in *Proc. ICASSP*, 2019, pp. 5746–5750.

[24] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE TASLP*, vol. 16, no. 1, pp. 229–238, 2008.